

**REKOMENDASI RUMAH MAKAN MALANG
MENGUNAKAN METODE *FUZZY ANALYTICAL HIERARCHY*
PROCESS DAN *TECHNIQUE FOR ORDER PREFERENCE BY*
*SIMILARITY TO IDEAL SOLUTION***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Mohammad Toriq
NIM : 125150200111102



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

REKOMENDASI RUMAH MAKAN MALANG MENGGUNAKAN METODE FUZZY
ANALYTICAL HIERARCHY PROCESS DAN TECHNIQUE FOR ORDER PREFERENCE BY
SIMILARITY TO IDEAL SOLUTION
SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

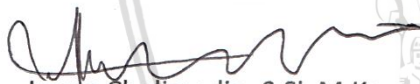
Disusun Oleh:
Mohammad Toriq
NIM: 125150200111102

Skripsi ini telah diuji dan dinyatakan lulus pada
8 Oktober 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II


Imam Cholissodin, S.Si, M.Kom
NIK: 201201 850719 1 001


Putra Pandu Adikaka, S.Kom, M.Kom
NIP. 19850725 200812 1 002

Mengetahui

Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D

NIP: 19710518 200312 1 001 

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata di dalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 18 Juli 2018



Mohammad Toriq
NIM: 125150200111102

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT atas anugerah serta limpahan rahmat-Nya, sehingga penulis dapat menyelesaikan skripsi ini. Skripsi ini disusun sebagai syarat memperoleh gelar sarjana pada Program Studi Teknik Informatika, Jurusan Teknik Informatika, Fakultas Ilmu Komputer Universitas Brawijaya.

Dalam proses penyelesaian skripsi ini penulis mendapatkan banyak bantuan, baik bantuan moral maupun materiil dari berbagai pihak. Oleh karena itu, penulis mengucapkan banyak terimakasih kepada:

1. Imam Cholissodin, S.Si, M.Kom selaku Pembimbing I yang telah memberikan bimbingan, arahan, ilmu, dan masukan dalam penyelesaian skripsi ini.
2. Putra Pandu Adikara, S.Kom, M.Kom, selaku Pembimbing II yang juga telah memberikan bimbingan, arahan, ilmu, dan masukan dalam penyelesaian skripsi ini.
3. Bapak dan Ibu dosen yang telah mendidik dan memberikan ilmu selama Penulis menempuh pendidikan di Fakultas Ilmu Komputer Universitas Brawijaya.
4. Segenap karyawan di Fakultas Ilmu Komputer Universitas Brawijaya yang membantu Penulis dalam pelaksanaan skripsi ini.
5. Kedua Orang Tua yaitu Bapak saya H. M Tamim dan Ibu saya Hj. Umaha dan adik-adikku yaitu Ahmad Badar Dwi Chaya dan Sofia Lababa beserta keluarga besar yang telah mendukung penulis dengan segala usahanya, mulai dari doa, materi, dukungan moral, semangat hidup, dan tauladan yang semata-mata untuk keberhasilan penulis.
6. Semua pihak yang tidak dapat disebutkan satu per satu, yang telah membantu dan terlibat baik secara langsung maupun tidak langsung dalam penulisan skripsi ini.

Penulis menyadari bahwa skripsi ini tidak lepas dari kesalahan dan kekurangan. Oleh karena itu, Penulis bersedia menerima kritik dan saran yang membangun untuk memperbaiki diri. Penulis berharap semoga skripsi ini dapat memberi manfaat.

Malang, 18 Juli 2018

Penulis

Toriq031194@gmail.com

ABSTRAK

Indonesia merupakan salah satu negara dengan jumlah penduduk besar mengalami peningkatan tiap tahun dalam bisnis kuliner, Terutama di kota-kota besar. Peningkatan rumah makanlah yang menyebabkan kebingungan dan membutuhkan waktu yang lama dalam memilih rumah makan. Maka dibutuhkan sistem rekomendasi rumah makan yang menggunakan kriteria-kriteria tertentu dalam penentuan perekomendasiannya. Permasalahan ini dapat diselesaikan dengan menggunakan metode *Fuzzy Analytical Hierarchy Process dan Technique for Order Preference by Similarity to Ideal Solution (F-AHP dan TOPSIS)*. Kriteria-kriteria yang digunakan adalah jumlah menu makan, rating rumah makan, harga menu makanan, jarak rumah makan dan lama waktu buka. Proses jalannya metode ini terbagi menjadi 2 tahapan: Tahapan pertama FAHP yakni dimulai matriks perbandingan kriteria, normalisasi matriks perbandingan kriteria, vektor bobot, bobot prioritas, *consistency ratio*, konversi matriks dengan TFN, Matriks sintesis *fuzzy*, vektor dan ordinat defuzzifikasi dan normalisasi vektor *fuzzy* atau dikenal sebagai bobot kriteria. Tahap kedua yakni TOPSIS dimulai dari pembuatan matriks keputusan, normalisasi matriks keputusan, matriks normalisasi terbobot, pencarian solusi ideal positif-negatif, pencarian jarak solusi ideal positif-negatif dan nilai *preference*. Hasil dari nilai *preference* akan dilakukan pengurutan untuk menghasilkan peringkat rumah makan yang direkomendasikan. Dalam penelitian ini melibatkan 3 orang pelanggan yang pernah mengunjungi rumah makan. Para pelanggan akan diminta perbandingan dari kriteria-kriteria rekomendasi rumah makan serta pemeringkatan rumah makan yang pernah dikunjungi oleh pelanggan. Pengujian penelitian ini menggunakan metode uji korelasi Spearman dalam penentuan hubungan kedekatan hasil pemeringkatan sistem terhadap pemeringkatan pakar secara langsung oleh masing-masing pelanggan. Hasil dari pengujian didapat bahwa tingkat akurasi dari pemeringkatan sistem terhadap masing-masing pelanggan rendah yakni 0,3352, -0,1538 dan ke tiga -0,3205 menunjukkan rendahnya tingkat akurasi hasil pemeringkatan sistem. Hal ini diperkuat dengan terdapat nilai negatif (-) yang menyatakan hubungan antar kedua variabel perbandingan sistem dengan perbandingan user berbanding terbalik.

Kata Kunci : rekomendasi rumah makan, *Fuzzy-AHP, F-AHP, TOPSIS*, uji korelasi Spearman.

ABSTRACT

Indonesia is one of the countries with large population increasing every year in culinary business, Especially in big cities. Increased restaurant that causes confusion and takes a long time in choosing a restaurant. Then a restaurant recommendation system that uses certain criteria in the recommendation recommender. This problem can be solved by using Fuzzy Analytical Hierarchy Process method and Technique for Order Preference by Similarity to Ideal Solution (F-AHP and TOPSIS). The criteria used are the number of food menu, restaurant rating, food menu price, restaurant distance and length of opening time. The process of this method is divided into two stages: The first stage of FAHP is the matrix of comparison of criteria, the normalization of the matrix of criteria comparison, weight vector, priority weight, consistency ratio, matrix conversion with TFN, fuzzy synthesis matrix, vector and ordinate defuzzification and normalization of fuzzy or vector known as the weighting criteria. The second stage of TOPSIS starts from making the decision matrix, the normalization of the decision matrix, the weighted normalization matrix, the search for the positive-negative ideal solution, the search for the positive-negative ideal solution distance and the preference value. The result of the preference value will be sequenced to produce the recommended restaurant rating. In this study involving 3 customers who had visited the restaurant. Customers will be asked for a comparison of restaurant recommendation criteria as well as ratings of restaurants that have been visited by customers. The research used Spearman correlation test method in determining the correlation relationship between system ranking result and direct ranking by each expert. The result of the test shows that the accuracy level of the system rating to each customer is low, that is 0,3352, -0,1538 and to -0,3 205 indicates the low level of accuracy of the rating result of the system. This is reinforced by having a negative (-) value that states the relationship between the two system ranking variables with user rankings is inversely proportional.

Keywords: recommendation restaurant, Fuzzy-AHP, F-AHP, TOPSIS, Spearman correlation Tersting.

DAFTAR ISI

| | |
|---|-----|
| PENGESAHAN | ii |
| PERNYATAAN ORISINALITAS | iii |
| KATA PENGANTAR..... | iv |
| ABSTRAK..... | v |
| ABSTRACT | vi |
| DAFTAR ISI | vii |
| DAFTAR TABEL..... | xi |
| DAFTAR GAMBAR..... | xii |
| DAFTAR KODE PROGRAM | xiv |
| DAFTAR LAMPIRAN | xv |
| BAB 1 PENDAHULUAN..... | 1 |
| 1.1 Latar belakang..... | 1 |
| 1.2 Rumusan masalah..... | 2 |
| 1.3 Tujuan | 2 |
| 1.4 Manfaat..... | 2 |
| 1.5 Batasan masalah | 2 |
| 1.6 Sistematika pembahasan..... | 3 |
| BAB 2 LANDASAN KEPUSTAKAAN | 4 |
| 2.1 Kajian Pustaka | 4 |
| 2.2 <i>Analytical Hierarchy Process (AHP)</i> | 5 |
| 2.3 <i>Fuzzy Analytical Hierarchy Process (FAHP)</i> | 9 |
| 2.4 <i>Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)</i> | 11 |
| 2.5 Korelasi Spearman (<i>Spearman Corelation</i>)..... | 12 |
| BAB 3 METODOLOGI | 14 |
| 3.1 Studi Kepustakaan | 14 |
| 3.2 Tipe Penelitian | 14 |
| 3.3 Partisipan Penelitian | 14 |
| 3.4 Lokasi Penelitian | 14 |
| 3.5 Pengumpulan Data | 14 |
| 3.6 Perancangan Sistem..... | 15 |

| | |
|---|----|
| 3.7 Implementasi | 15 |
| 3.8 Pengujian Sistem..... | 15 |
| 3.9 Kesimpulan dan Saran | 15 |
| BAB 4 PERANCANGAN DAN IMPLEMENTASI | 16 |
| 4.1 Formulasi Permasalahan..... | 16 |
| 4.2 Siklus algoritme FAHP-TOPSIS..... | 17 |
| 4.2.1 Matriks Perbandingan Kriteria | 19 |
| 4.2.2 Normalisasi Matriks Perbandingan Kriteria | 19 |
| 4.2.3 Vektor Bobot | 21 |
| 4.2.4 Nilai Prioritas | 22 |
| 4.2.5 <i>Consistency Ratio</i> | 22 |
| 4.2.6 Matriks Perbandingan Kriteria Dengan Skala TFN | 24 |
| 4.2.7 Matriks Sintesis <i>Fuzzy</i> | 25 |
| 4.2.8 Vektor dan Ordinat Defuzzifikasi | 27 |
| 4.2.9 Normalisasi Vektor <i>Fuzzy</i> | 28 |
| 4.2.10 Matriks keputusan | 29 |
| 4.2.11 Normalisasi matriks keputusan..... | 32 |
| 4.2.12 Matriks Normalisasi Terbobot | 33 |
| 4.2.13 Solusi Ideal Positif dan Solusi Ideal Negatif | 35 |
| 4.2.14 Jarak Solusi Ideal Positif dan Jarak Solusi Ideal Negatif | 36 |
| 4.2.15 Nilai <i>Preference</i> | 37 |
| 4.3 Siklus Penyelesaian Rekomendasi Rumah Makan Menggunakan Algoritme FAHP-TOPSIS | 38 |
| 4.3.1 Matriks Perbandingan Kriteria | 38 |
| 4.3.2 Normalisasi Matriks Perbandingan Kriteria | 39 |
| 4.3.3 Vektor Bobot | 40 |
| 4.3.4 Nilai Prioritas | 40 |
| 4.3.5 <i>Consistency Ratio</i> | 41 |
| 4.3.6 Matriks Perbandingan Kriteria dengan Skala TFN | 41 |
| 4.3.7 Matriks Sintesis <i>Fuzzy</i> | 43 |
| 4.3.8 Vektor dan Ordinat Defuzzifikasi | 44 |
| 4.3.9 Normalisasi vektor <i>defuzzy</i> | 46 |
| 4.3.10 Matriks Keputusan | 46 |

| | |
|--|----|
| 4.3.11 Normalisasi Matriks Keputusan | 48 |
| 4.3.12 Matriks Normalisasi Terbobot | 48 |
| 4.3.13 Solusi Ideal Positif dan Solusi Ideal Negatif | 49 |
| 4.3.14 Jarak Solusi Ideal Positif dan Jarak Solusi Ideal Negatif | 50 |
| 4.3.15 Nilai <i>Preference</i> | 51 |
| 4.4 Perancangan Pengujian | 52 |
| 4.5 Perancangan Antarmuka | 53 |
| 4.5.1 Perancangan Antarmuka Menu Input Data Pelanggan | 53 |
| 4.5.2 Perancangan Antarmuka Menu Input Data Rumah Makan..... | 54 |
| 4.5.3 Perancangan Antarmuka Menu FAHP,TOPSIS, Rekomendasi dan Pengujian..... | 55 |
| 4.6 Implementasi algoritme FAHP-TOPSIS | 56 |
| 4.6.1 Implementasi Matriks Perbandingan Kriteria..... | 56 |
| 4.6.2 Implementasi Normalisasi Matriks Perbandingan Kriteria | 57 |
| 4.6.3 Implementasi Vektor Bobot | 57 |
| 4.6.4 Implementasi Nilai Prioritas | 58 |
| 4.6.5 Implementasi Pengecekan Konsistensi | 59 |
| 4.6.6 Implementasi Konversi Matriks Kriteria skala TFN | 59 |
| 4.6.7 Implementasi Sintesis <i>Fuzzy</i> | 61 |
| 4.6.8 Implementasi Vektor dan Ordinat Defuzzifikasi | 62 |
| 4.6.9 Implementasi Normalisasi Vektor defuzzifikasi | 63 |
| 4.6.10 Implementasi Matriks Keputusan | 64 |
| 4.6.11 Implementasi Normalisasi Matriks Keputusan | 66 |
| 4.6.12 Implementasi Normalisasi Terbobot..... | 66 |
| 4.6.13 Implementasi Solusi Ideal Positif & Ideal Negatif | 67 |
| 4.6.14 Implementasi Jarak Solusi Ideal Positif & Jarak Ideal Negatif... | 68 |
| 4.6.15 Implementasi Nilai <i>Preference</i> | 69 |
| 4.7 Implementasi Antarmuka | 69 |
| BAB 5 PENGUJIAN DAN ANALISIS..... | 76 |
| 5.1 Pengujian Korelasi Spearman | 76 |
| 5.2 Analisis Korelasi Spearman | 78 |
| BAB 6 PENUTUP | 80 |
| 6.1 Kesimpulan..... | 80 |

| | |
|--|----|
| 6.2 Saran | 80 |
| DAFTAR PUSTAKA..... | 82 |
| LAMPIRAN MATRIKS KRITERIA PELANGGAN | 84 |
| LAMPIRAN DATA RUMAH MAKAN PELANGGAN | 85 |
| LAMPIRAN PEMERINGKATAN RUMAH MAKAN PELANGGAN | 87 |



DAFTAR TABEL

| | |
|--|----|
| Tabel 2.1 Kajian Pustaka | 4 |
| Tabel 2.2 Nilai skala perbandingan berpasangan | 6 |
| Tabel 2.3 Nilai <i>Ratio Index</i> | 8 |
| Tabel 2.4 Skala <i>Triangular Fuzzy Number</i> (TFN). | 9 |
| Tabel 2.5 Koefisien korelasi D.A. de Vaus | 13 |
| Tabel 4.1 Data rumah makan di kota Malang..... | 16 |
| Tabel 4.2 Perbandingan kriteria | 38 |
| Tabel 4.3 Normalisasi matriks perbandingan kriteria | 39 |
| Tabel 4.4 Bobot kriteria awal | 39 |
| Tabel 4.5 Vektor bobot | 40 |
| Tabel 4.6 Nilai prioritas kriteria..... | 40 |
| Tabel 4.7 Konversi matriks dengan skala TFN..... | 42 |
| Tabel 4.8 Penjumlahan pada kolom <i>lower</i> , <i>middle</i> dan <i>upper</i> | 43 |
| Tabel 4.9 Matriks sintesis <i>fuzzy</i> | 44 |
| Tabel 4.10 Nilai vektor defuzzifikasi..... | 45 |
| Tabel 4.11 Nilai ordinat defuzzifikasi | 45 |
| Tabel 4.12 Bobot kriteria..... | 46 |
| Tabel 4.13 Konversi data | 46 |
| Tabel 4.14 Matriks keputusan..... | 47 |
| Tabel 4.15 Matriks normalisasi keputusan | 48 |
| Tabel 4.16 Matriks normalisasi terbobot..... | 49 |
| Tabel 4.17 Matriks ideal positif-negatif | 49 |
| Tabel 4.18 Matriks jarak ideal positif-negatif | 50 |
| Tabel 4.19 Nilai <i>preference</i> | 51 |
| Tabel 4.20 Pengurutan nilai <i>preference</i> | 51 |
| Tabel 4.21 Hasil uji pelanggan pertama | 52 |
| Tabel 5.1 Hasil uji pelanggan pertama | 76 |
| Tabel 5.2 Hasil uji pelanggan kedua | 77 |
| Tabel 5.3 Hasil uji pelanggan ketiga | 77 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Susunan hierarki AHP | 6 |
| Gambar 4.1 Hierarki permasalahan | 17 |
| Gambar 4.2 <i>Flowchart</i> siklus FAHP-TOPSIS..... | 18 |
| Gambar 4.3 <i>Flowchart</i> matriks perbandingan kriteria | 19 |
| Gambar 4.4 <i>Flowchart</i> normalisasi matriks perbandingan kriteria | 20 |
| Gambar 4.5 <i>Flowchart</i> perhitungan bobot prioritas..... | 21 |
| Gambar 4.6 <i>Flowchart</i> perhitungan bobot prioritas..... | 22 |
| Gambar 4.7 <i>Flowchart consistency ratio</i> | 23 |
| Gambar 4.8 <i>Flowchart</i> proses matriks skala TFN..... | 24 |
| Gambar 4.9 <i>Flowchart</i> proses matriks sintesis fuzzy | 26 |
| Gambar 4.10 <i>Flowchart</i> Proses perhitungan vektor dan ordinat defuzzifikasi | 28 |
| Gambar 4.11 <i>Flowchart</i> normalisasi vektor fuzzy..... | 29 |
| Gambar 4.12 <i>Flowchart</i> normalisasi vektor fuzzy..... | 31 |
| Gambar 4.13 <i>Flowchart</i> matriks keputusan ternormalisasi | 33 |
| Gambar 4.14 <i>Flowchart</i> matriks normalisasi terbobot..... | 34 |
| Gambar 4.15 <i>Flowchart</i> solusi ideal positif dan solusi ideal negatif..... | 35 |
| Gambar 4.16 <i>Flowchart</i> jarak ideal positif dan jarak ideal negatif..... | 37 |
| Gambar 4.17 <i>Flowchart</i> pencarian nilai <i>preference</i> | 38 |
| Gambar 4.18 Perancangan antarmuka menu input data pelanggan | 54 |
| Gambar 4.19 Perancangan antarmuka menu input data rumah makan..... | 54 |
| Gambar 4.20 Perancangan antarmuka menu fahp, topsis, hasil rekomendasi dan pengujian..... | 55 |
| Gambar 4.21 Halaman input data pelanggan bagian nama, lokasi, perbandingan kriteria | 70 |
| Gambar 4.22 Halaman Input Data Pelanggan Bagian Rating..... | 70 |
| Gambar 4.23 Halaman input data rumah makan | 71 |
| Gambar 4.24 Halaman proses FAHP bagian matriks kriteria, normalisasi dan bobot kriteria awal..... | 71 |
| Gambar 4.25 Halaman proses FAHP bagian vektor bobot, nilai prioritas hingga cr | 72 |
| Gambar 4.26 Halaman proses FAHP bagian konversi TFN dan sintesis fuzzy | 72 |

| | |
|--|----|
| Gambar 4.27 Halaman proses FAHP bagian vektor dan ordinat defuzzifikasi hingga normalisasi | 73 |
| Gambar 4.28 Halaman proses TOPSIS bagian matriks keputusan dan normalisasi | 74 |
| Gambar 4.29 Halaman proses TOPSIS bagian normalisasi terbobot, solusi ideal positif-negatif dan jarak ideal positif-negatif..... | 74 |
| Gambar 4.30 Halaman proses TOPSIS bagian nilai presensi dan pemeringkatan rumah makan | 75 |



DAFTAR KODE PROGRAM

| | |
|---|----|
| Kode Program 4.1 Matriks kriteria perbandingan | 56 |
| Kode Program 4.2 Normalisasi matriks kriteria perbandingan..... | 57 |
| Kode Program 4.3 Vektor bobot | 58 |
| Kode Program 4.4 Nilai prioritas | 58 |
| Kode Program 4.5 Pengecekan konsistensi | 59 |
| Kode Program 4.6 Konversi matriks kriteria skala TFN..... | 60 |
| Kode Program 4.7 Sintesis <i>fuzzy</i> | 61 |
| Kode Program 4.8 Vektor dan ordinat defuzzifikasi | 63 |
| Kode Program 4.9 Normalisasi vektor defuzzifikasi..... | 63 |
| Kode Program 4.10 Matriks keputusan | 65 |
| Kode Program 4.11 Normalisasi matriks keputusan..... | 66 |
| Kode Program 4.12 Normalisasi terbobot | 67 |
| Kode Program 4.13 Solusi ideal positif & ideal negatif..... | 68 |
| Kode Program 4.14 Jarak solusi ideal positif & ideal negatif..... | 68 |
| Kode Program 4.15 Nilai <i>preference</i> | 69 |

DAFTAR LAMPIRAN

| | |
|---|----|
| A.1. Matriks Kriteria Perbandingan Pelanggan 1..... | 84 |
| A.2. Matriks Kriteria Perbandingan Pelanggan 2..... | 84 |
| A.3. Matriks Kriteria Perbandingan Pelanggan 3..... | 84 |
| B.1. Data Rumah Makan Pelanggan 1..... | 85 |
| B.2. Data Rumah Makan Pelanggan 2..... | 85 |
| B.3. Data Rumah Makan Pelanggan 3..... | 86 |
| C.1. Data Pemeringkatan Rumah Makan Pelanggan 1 | 87 |
| C.2. Data Pemeringkatan Rumah Makan Pelanggan 2 | 87 |
| C.3. Data Pemeringkatan Rumah Makan Pelanggan 3 | 88 |



BAB 1 PENDAHULUAN

1.1 Latar belakang

Makanan merupakan salah satu dari 3 kebutuhan *primer* yang harus dimiliki oleh setiap manusia. Kebutuhan akan makanan meningkat seiring dengan bertambahnya jumlah masyarakat suatu wilayah dari tahun ke tahun (Pratama, et al., 2017). Ketertarikan masyarakat akan makanan dipengaruhi faktor kegiatan primer masing-masing orang dalam memenuhi kebutuhan pangan, sehingga ke manapun seseorang pergi ke suatu tempat, pasti akan mencari makanan (Primasari & Siswojo, 2012).

Indonesia merupakan salah satu negara dengan jumlah penduduk besar mengalami peningkatan tiap tahun dalam bisnis kuliner, Terutama di kota-kota besar. Kota Malang Pada tahun 2014, memiliki 549 rumah makan, 34 restoran dan 45 Café, jumlah ini mengalami peningkatan dari jumlah restoran dan rumah makan di tahun 2013 yakni 466 rumah makan, 45 restoran, dan 23 Café (Triana, 2015). Peningkatan rumah makanlah yang menyebabkan kebingungan dan membutuhkan waktu yang lama dalam memilih rumah makan. Faktor lain yang menyebabkan kebingungan dalam menentukan rumah makan adalah kriteria-kriteria yang masing-masing orang dengan yang lainnya berbeda terkait dengan rumah makan yang akan dipilih atau menu makanan dan minuman yang dimiliki setiap rumah makan. Sistem rekomendasi rumah makan sangat dibutuhkan dalam memesan makanan dan minuman dari rumah makan sesuai kriteria masing-masing untuk mempersingkat waktu dalam memilih dari banyaknya rumah makan di suatu wilayah khususnya kota Malang.

Pada penelitian sebelumnya dilakukan uji coba akurasi sebanyak 30 kali dengan menggunakan 5 data alternatif makanan yang berbeda-beda menggunakan data sumber dari 35 tempat kuliner dan 244 daftar menu makanan di Kota Malang. Hasil keluaran yang didapat adalah antara menggunakan *Analytical Hierarchy Process dan Technique for Order Preference by Similarity to Ideal Solution* terhadap pilihan pakar ahli kuliner yakni 40% (Nurrachman, 2016). Hasil keluaran yang didapat meningkat 10% dari penelitian yang terdahulunya dengan tingkat akurasi hanya 30% (Pinandito, Wulandari & Dewi, 2015). Adapun penelitian lainnya yang bertujuan untuk mengetahui bagaimana hasil alternatif metode rekomendasi kuliner menggunakan metode *Fuzzy Analytical Hierarchy Process* terhadap method AHP tradisional. Percobaan dilakukan sama halnya dengan penelitian sebelumnya yakni uji coba akurasi sebanyak 30 kali dengan menggunakan 5 data alternatif makanan yang berbeda-beda menggunakan data sumber dari 35 tempat kuliner dan 244 daftar menu makanan di Kota Malang. Hasil dari penelitian ini adalah metode F-AHP memiliki hasil pembobotan yang lebih baik daripada metode AHP tradisional 66.67% (Pinandito, Ananta, Brata &

Fanani, 2015). F-AHP terhadap pilihan pakar ahli kuliner. Hasil keluaran yang didapat meningkat 36.67% dari penelitian yang terdahulunya dengan tingkat akurasi hanya 30% (Pinandito, Wulandari & Dewi, 2015).

Dari beberapa penelitan di atas terdapat peningkatan akurasi pada hasil dari 2 penelitian tersebut terhadap penelitian sebelumnya. Penulis berencana mengangkat sebuah topik penelitian yang berjudul **“Rekomendasi rumah makan Malang menggunakan metode fuzzy analytic hierarchy process dan technique for order preference by similarity to ideal solution”** Tujuan utama dari penelitian ini selain memberikan rekomendasi makanan yang akan dipilih oleh pengguna sistem sesuai dengan kriteria yang ditentukan pengguna sendiri. Kriteria yang akan digunakan dalam penelitian ini adalah jumlah menu, jumlah porsi, harga menu, jarak serta waktu buka.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah dijelaskan di atas, dapat dirumuskan beberapa masalah sebagai berikut.

1. Bagaimana proses pemberian rekomendasi rumah makan menggunakan metode *Fuzzy Analytical Hierarchy Process Dan Technique For Order Preference By Similarity To Ideal Solution*?
2. Bagaimana hubungan kedekatan hasil pemeringkatan rekomendasi rumah makan dari metode *Fuzzy Analytical Hierarchy Process Dan Technique For Order Preference By Similarity To Ideal Solution* terhadap hasil pemeringkatan pelanggan?

1.3 Tujuan

Tujuan yang ingin dicapai dari pembuatan skripsi ini yaitu:

1. Menunjukkan proses pemberian rekomendasi rumah makan menggunakan metode *Fuzzy Analytical Hierarchy Process Dan Technique For Order Preference By Similarity To Ideal Solution*.
2. Mengetahui hubungan kedekatan hasil pemeringkatan rekomendasi rumah makan dari metode *Fuzzy Analytical Hierarchy Process Dan Technique For Order Preference By Similarity To Ideal Solution* terhadap hasil pemeringkatan pelanggan.

1.4 Manfaat

Penulisan skripsi ini diharapkan dapat bermanfaat bagi calon pembeli dalam memilih rumah makan yang akan dikunjungi.

1.5 Batasan masalah

Batasan dalam penelitian ini, agar penelitian ini dapat terfokus adalah:

1. Data yang digunakan dalam penelitian ini adalah data rumah makan di Kota Malang yang telah dikunjungi.
2. Kriteria transaksi yang digunakan untuk menentukan hasil rekomendasi adalah jumlah menu, jumlah porsi, harga menu, jarak serta waktu buka.

1.6 Sistematika pembahasan

1. BAB I. PENDAHULUAN

Bab ini memuat latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat serta sistematika penulisan.

2. BAB II. LANDASAN KEPUSTAKAAN

Pada bab ini berisi tentang penjelasan dan dasar-dasar teori tentang pengetahuan dasar yang relevan dengan yang skripsi diteliti.

3. BAB III. METODOLOGI

Pada bab ini berisi tentang penjelasan studi literatur, pengumpulan data, analisis dan perancangan, implementasi, pengujian, serta evaluasi dan hasil analisis.

4. BAB IV. PERANCANGAN DAN IMPLEMENTASI

Pada bab perancangan membahas mengenai beberapa perihal yang dilakukan untuk menyelesaikan permasalahan di dalam skripsi yang diteliti yang terdiri dari formulasi permasalahan, siklus algoritme, siklus penyelesaian masalah, perancangan pengujian, dan perancangan antarmuka.

5. BAB V. PENGUJIAN DAN PEMBAHASAN

Bab pengujian dan pembahasan menguraikan mengenai pengujian beberapa parameter. Adapun analisis hasil yang membahas mengenai pengujian secara keseluruhan.

6. BAB VI. PENUTUP

Pada bab ini berisi ringkasan dan pencapaian hasil dan menjawab pertanyaan dari rumusan masalah. Serta saran yang dapat diberikan untuk pengembangan lebih lanjut.

BAB 2 LANDASAN KEPUSTAKAAN

Landasan kepastakaan berisi uraian dan pembahasan tentang teori, konsep, model, metode, atau sistem dari literatur ilmiah, yang berkaitan dengan tema, masalah, atau pertanyaan penelitian penulis. Dalam landasan kepastakaan terdapat landasan teori dari berbagai sumber pustaka yang terkait dengan teori dan metode yang digunakan dalam penelitian.

2.1 Kajian Pustaka

Kajian pustaka menjelaskan secara umum penelitian penelitian sebelumnya terkait dengan penelitian penulis dijelaskan pada Tabel 2.1.

Tabel 2.1 Kajian Pustaka

| No | Judul | Obyek | Metode | Hasil |
|----|--|---------------------|---|---|
| 1. | <i>Culinary Recommendation Systems Using Analytical Hierarchy Process on Google Android Platform</i> | Kuliner Kota Malang | <i>Analytical Hierarchy Process</i> | Penelitian ini menyelesaikan permasalahan rekomendasi kuliner di kota Malang dengan menggunakan kriteria jarak rumah makan, harga makanan dan peringkat makanan. Hasil dari penelitian ini adalah akurasi perbandingan hasil keluaran antara sistem dengan ahli pakar sebesar 30%. |
| 2. | <i>Alternatives weighting in analytic hierarchy process of mobile culinary recommendation system using fuzzy</i> | Kuliner Kota Malang | <i>Fuzzy Analytical Hierarchy Process</i> | Penelitian ini menyelesaikan permasalahan rekomendasi kuliner di kota Malang dengan menggunakan kriteria jarak rumah makan, harga makanan dan peringkat makanan. Hasil dari penelitian ini adalah akurasi perbandingan hasil keluaran antara sistem dengan ahli pakar sebesar 66.67%. |
| 3. | Implementasi Metode AHP-Topsis pada Sistem Rekomendasi Kuliner | Kuliner Kota Malang | <i>Analytical Hierarchy Process</i> dan <i>Technique for Order Preference by Similarity to Ideal Solution</i> | Penelitian ini menyelesaikan permasalahan rekomendasi kuliner di kota Malang dengan menggunakan kriteria jarak rumah makan, harga makanan dan peringkat makanan. Hasil dari penelitian ini adalah akurasi perbandingan hasil keluaran antara sistem dengan ahli pakar sebesar 40%. |

Beberapa penelitian terdahulu yang berkaitan dengan penelitian penulis antara lain:

Penelitian Pertama adalah dilakukan oleh Pinandito, Wulandari dan Dewi (2015), bertujuan untuk mengetahui bagaimana tingkat akurasi antara hasil dari sistem yang menggunakan metode AHP dengan ahli pakar. Hasil yang diperoleh adalah menunjukkan tingkat akurasi rekomendasi sistem sebesar 30% terhadap hasil rekomendasi ahli pakar. Pengujian itu dilakukan sebanyak 30 kali dengan menggunakan 5 data alternatif makanan yang berbeda-beda menggunakan data sumber dari 35 tempat kuliner dan 244 daftar menu makanan di Kota Malang

Penelitian Kedua adalah dilakukan oleh Pinandito, Ananta, Brata dan Fanani (2015), bertujuan untuk mengetahui bagaimana hasil alternatif metode rekomendasi kuliner menggunakan F-AHP terhadap method AHP tradisional. Hasil yang diperoleh adalah menunjukkan tingkat akurasi rekomendasi sistem sebesar 66.67% terhadap hasil rekomendasi ahli pakar. Pengujian itu dilakukan sebanyak 30 kali dengan menggunakan 5 data alternatif makanan yang berbeda-beda menggunakan data sumber dari 35 tempat kuliner dan 244 daftar menu makanan di Kota Malang

Penelitian Ketiga (Nurrachman, 2016), adalah mengimplementasikan Metode AHP-TOPSIS pada aplikasi mobile untuk sistem rekomendasi kuliner. Penelitian ini bertujuan untuk memberikan rekomendasi kuliner kepada pengguna sesuai kriteria pengguna. Ada 2 tahapan yakni pebobotan kriteria yang dilakukan dengan metode AHP. Hasil dari pebobotan tersebut akan digunakan untuk pemeringkatan rekomendasi menggunakan metode TOPSIS. Hasil yang diperoleh adalah menunjukkan tingkat akurasi rekomendasi sistem sebesar 40% terhadap hasil rekomendasi ahli pakar. Pengujian itu dilakukan sebanyak 30 kali dengan menggunakan 5 data alternatif makanan yang berbeda-beda menggunakan data sumber dari 35 tempat kuliner dan 244 daftar menu makanan di Kota Malang.

Berdasarkan beberapa penelitian yang telah dilakukan sebelumnya, penulis menyelesaikan masalah rekomendasi tempat makan menggunakan metode gabungan antara F-AHP dengan TOPSIS. Adapun kriteria yang digunakan dalam penelitian ini adalah Kriteria yang akan digunakan dalam penelitian ini adalah jumlah menu, jumlah porsi, harga menu, jarak serta waktu buka.

2.2 Analytical Hierarchy Process (AHP)

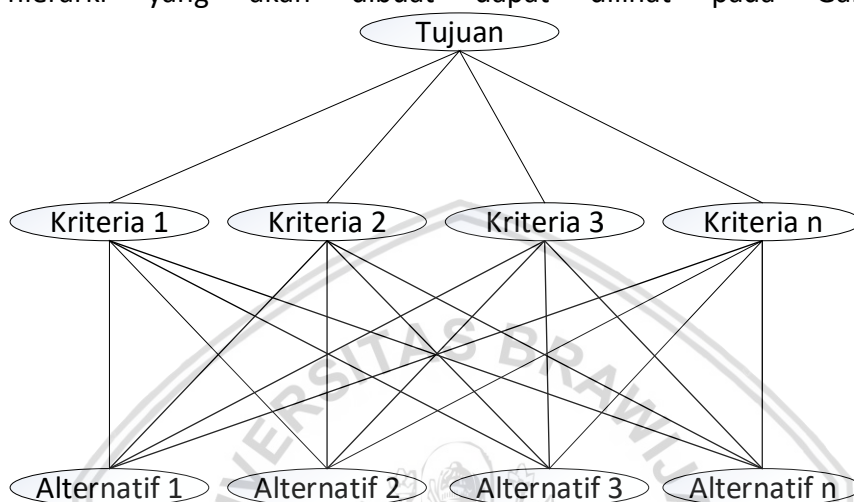
Analytical Hierarchy Process (AHP) merupakan metode yang dikembangkan oleh Prof. Thomas Lorie Saaty pada awal tahun 1970. Metode ini merupakan sebuah hierarki fungsional dengan persepsi manusia sebagai inputannya. Metode ini awalnya digunakan oleh Prof. Thomas Lorie Saaty untuk mencari urutan prioritas dari berbagai alternatif dalam pemecahan suatu masalah (Purnomo, Sihwi, & Anggrainingsih, 2013).

AHP dapat menyelesaikan beberapa permasalahan yang memiliki kriteria yang banyak. Adapaun contoh dari beberapa masalah adalah penentuan prioritas

pelanggan, pemilihan kebijakan terbaik setelah menemukan satu set alternatif, dan berbagai permasalahan lainnya yang memiliki banyak kriteria dalam penentuan pengurutan prioritas alternatif penyelesaian masalah.

Tahapan-tahapan dalam pengerjaan metode AHP adalah sebagai berikut.

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan, lalu menyusun hierarki permasalahan yang dihadapi. Berikut contoh susunan hierarki yang akan dibuat dapat dilihat pada Gambar 2.1.



Gambar 2.1 Susunan hierarki AHP

2. Menentukan prioritas elemen dengan membuat matriks perbandingan berpasangan yang diisi dengan bilangan untuk mempresentasikan kepentingan relatif antar elemen dengan menggunakan skala ada Tabel 2.2.

Tabel 2.2 Nilai skala perbandingan berpasangan

| Intensitas Kepentingan | Definisi | Keterangan |
|------------------------|-----------------------|---|
| 1 | Sama Penting | Kedua elemen mempunyai pengaruh sama. |
| 3 | Sedikit Lebih penting | Satu elemen sedikit lebih penting daripada lainnya. |
| 5 | Lebih Penting | Satu elemen lebih penting daripada lainnya. |
| 7 | Sangat Penting | Satu elemen lebih mutlak penting daripada lainnya. |
| 9 | Mutlak lebih penting | Satu elemen mutlak penting daripada lainnya . |
| 2,4,6,8 | Nilai Tengah | Nilai antara dua nilai yang berdekatan. |
| Respirokal | Kebalikannya | Jika elemen i memiliki nilai di atas maka elemen j memiliki nilai berbanding terbalik terhadap elemen i |

Sumber (Purnomo, Sihwi, & Anggrainingsih, 2013)

3. Normalisasi matriks perbandingan berpasangan, dengan cara:

- a. Membagi setiap nilai pada matriks dengan hasil penjumlahan dari nilai setiap kolom pada matriks perbandingan berpasangan dengan Persamaan 2.1

$$nm_j = \frac{x_{ij}}{\sum_{i=0}^n x_{ij}} \quad (2.1)$$

nm_{ij} = hasil normalisasi *cell*

n = banyak elemen

x_{ij} = nilai tiap *cell*

i = indeks baris

j = indeks kolom

4. Menghitung nilai bobot prioritas dengan cara menjumlahkan nilai-nilai yang telah di normalisasi setiap baris dan membagi hasil jumlahnya dengan banyaknya kriteria sesuai dengan Persamaan 2.2

$$bp_i = \frac{\sum_{j=0}^n x_{ij}}{n} \quad (2.2)$$

bp_i = bobot prioritas baris i

n = banyak elemen

x_{ij} = nilai setiap *cell*

i = indeks baris

j = indeks kolom

5. Menghitung Eigen Maksimum dengan cara:

- a. Perkalian matriks antara matriks perbandingan berpasangan dengan matriks bobot prioritas dengan Persamaan 2.3

$$vb_i = \sum_{j=0}^n x_{ij} * bp_j \quad (2.3)$$

vb_i = vektor bobot baris i

n = banyak elemen

x_{ij} = nilai setiap *cell*

bp_i = bobot prioritas baris i

i = indeks baris

j = indeks kolom

- b. Membagi hasil dari vektor bobot dengan bobot prioritas yang berkaitan dengan Persamaan 2.4

$$p_i = \frac{vb_i}{bp_i} \quad (2.4)$$

p_i = nilai prioritas baris i

vb_i = vektor bobot baris i

bp_i = bobot prioritas baris i

i = indeks baris

- c. Menghitung nilai Eigen maksimum dengan menjumlahkan semua nilai prioritas kemudian dibagi dengan banyaknya elemen dengan Persamaan 2.5

$$\lambda_{maks} = \frac{\sum_{i=0}^n p_i}{n} \quad (2.5)$$

λ_{maks} = Eigen maksimum

p_i = nilai prioritas baris i

n = banyak elemen

6. Menghitung indeks konsistensi (*Consistency Index*) dengan menggunakan Persamaan 2.6

$$CI = \frac{\lambda_{maks} - n}{n - 1} \quad (2.6)$$

CI = *Consistency Index*

λ_{maks} = Eigen maksimum

n = banyak elemen

7. Menghitung *Consistency Ratio* dengan membagi nilai *Consistency Index* dengan *Ratio Index*. *Ratio Index* didapat dari Tabel 2.3.

Tabel 2.3 Nilai *Ratio Index*

| Urutan Matriks | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------------------|------|------|------|------|------|------|------|------|------|------|
| RI | 0,00 | 0,00 | 0,58 | 0,90 | 1,12 | 1,24 | 1,32 | 1,41 | 1,45 | 1,49 |

Persamaan yang digunakan adalah:

$$CR = \frac{CI}{RI} \quad (2.7)$$

CR = *Consistency Ratio*

CI = *Consistency Index*

RI = *Ratio Index*

8. Memeriksa konsistensi hierarki dengan cara apabila nilai CR yang didapat lebih dari 10% atau 0,1 maka nilai matriks perbandingan berpasangan harus diperbaiki kembali nilai matriks perbandingan kriteria. Dan jika nilai CR yang didapat kurang dari sama dengan 10% atau 0,1 maka perhitungan data konsisten dan benar.

2.3 Fuzzy Analytical Hierarchy Process (FAHP)

Fuzzy Analytical Hierarchy Process atau FAHP merupakan salah satu dari pengembangan model AHP yang dikembangkan oleh para peneliti dengan pendekatan konsep *Fuzzy* (Raharjo dkk, 2002). Metode F-AHP dipilih untuk menutupi kelemahan dari metode AHP tradisional dalam menangani permasalahan pada kriteria yang memiliki sifat subjektif lebih banyak (Chang, 1996). Chang (1996), mendefinisikan langkah-langkah dalam menggunakan metode F-AHP sebagai berikut.

Langkah pertama adalah membagi permasalahan yang akan diselesaikan ke dalam beberapa elemen. Elemen-elemen tersebut akan disusun ke dalam bentuk hierarki dengan urutan seperti pembuatan hierarki dalam metode AHP tradisional. Langkah berikutnya sama seperti method AHP tradisional yakni membuat matriks perbandingan berpasangan. Namun pada tahapan ini skala yang digunakan dari satty di konversi menggunakan skala *Triangular Fuzzy Number* (TFN) seperti Tabel 2.4.

Tabel 2.4 Skala *Triangular Fuzzy Number* (TFN).

| Intensitas Kepentingan AHP | Himpunan Linguistik | <i>Triangular Fuzzy Number</i> (TFN) | <i>Reciprocal</i> (Kebalikan) |
|----------------------------|--|--------------------------------------|-------------------------------|
| 1 | Perbandingan elemen yang sama | (1, 1, 1) | (1, 1, 1) |
| 2 | Pertengahan | (1/2, 1, 3/2) | (2/3, 1, 2) |
| 3 | Elemen satu cukup penting dari yang lainnya | (1, 3/2, 2) | (1/2, 2/3, 1) |
| 4 | Pertengahan | (3/2, 2, 5/2) | (2/5, 1/2, 2/3) |
| 5 | Elemen satu kuat pentingnya dari yang lain | (2, 5/2, 3) | (1/3, 2/5, 1/2) |
| 6 | Pertengahan | (5/2, 3, 7/2) | (2/7, 1/3, 2/5) |
| 7 | Elemen satu lebih kuat pentingnya dari yang lain | (3, 7/2, 4) | (1/4, 2/7, 1/3) |
| 8 | Pertengahan | (7/2, 4, 9/2) | (2/9, 1/4, 2/7) |
| 9 | Elemen satu mutlak lebih penting dari yang lainnya | (4, 9/2, 9/2) | (2/9, 2/9, 1/4) |

Langkah kedua adalah menentukan nilai sintesis fuzzy (S_i) prioritas dengan Persamaan 2.8.

$$S_i = \sum_{j=1}^m M_g^j * [\sum_{i=1}^n \sum_{j=1}^m M_{gi}^j]^{-1} \quad (2.8)$$

Dengan variable S_i sebagai nilai sintesis fuzzy, $\sum_{j=1}^m M_{gi}^j$ adalah penjumlahan dari nilai *cell* pada kolom yang dimulai dari kolom j di setiap baris matriks, i adalah indeks baris dan j adalah indeks kolom pada matriks perbandingan berpasangan yang telah dikonversi. M merupakan bilangan *triangular fuzzy*, m adalah jumlah kriteria dan g adalah parameter (l, m, u) pada matriks perbandingan berpasangan yang telah di konversi dengan skala *triangular fuzzy number*.

Dalam menentukan variabel $\sum_{j=1}^m M_{gi}^j$, dilakukan operasi penjumlahan untuk keseluruhan bilangan *triangular fuzzy* dalam matriks keputusan ($n \times m$), menggunakan Persamaan 2.9.

$$\sum_{j=1}^m M_g^j = (\sum_{j=1}^m l_j, \sum_{j=1}^m m_j, \sum_{j=1}^m u_j) \quad (2.9)$$

Pada Persamaan 2.9 ini menjelaskan bahwa variabel $\sum_{j=1}^m l_j$ adalah jumlah *cell* pada kolom pertama matriks (nilai lower), variabel $\sum_{j=1}^m m_j$ adalah jumlah *cell* pada kolom kedua matriks (median) dan variabel $\sum_{j=1}^m u_j$ adalah jumlah *cell* pada kolom ketiga matriks (upper). Dan untuk menghitung persamaan inversnya mengikuti Persamaan 2.10.

$$[\sum_{i=1}^n \sum_{j=1}^m M_{gi}^j]^{-1} = \left(\frac{1}{\sum_{i=1}^m u_i}, \frac{1}{\sum_{i=1}^m m_i}, \frac{1}{\sum_{i=1}^m l_i} \right) \quad (2.10)$$

Langkah Ketiga adalah Perbandingan tingkat kemungkinan antara bilangan fuzzy. Perbandingan ini digunakan untuk nilai bobot pada masing-masing kriteria yang memiliki bilangan *triangular fuzzy* $M = (l, m, u)$. Perbandingan 2 buah bilangan *triangular fuzzy* misal M_1 dan M_2 , dengan tingkat kemungkinan $M_2 \geq M_1$ dapat didefinisikan dengan Persamaan 2.11.

$$V(M_2 \geq M_1) = \sup[\min(\mu_{M_1}(x), \mu_{M_2}(y))] \quad (2.11)$$

Tingkat kemungkinan juga dapat di peroleh dengan Persamaan 2.12.

$$V(M_2 \geq M_1) = \begin{cases} 1; & \text{jika } m_2 \geq m_1 \\ 0; & \text{jika } l_1 \geq u_2 \\ \frac{l_1 - u_2}{(m_2 - u_2) - (m_1 - l_1)} & \text{untuk kondisi lain} \end{cases} \quad (2.12)$$

Langkah berikutnya adalah jika hasil kemungkinan masing 2 kriteria lebih dari satu dengan jumlah kriteria minimal adalah 3 maka dapat menggunakan Persamaan 2.13.

$$\begin{aligned} V(M \geq M_1, M_2, \dots, M_k) \\ = V[(M \geq M_1), \text{ dan }, (M \geq M_2), \text{ dan }, \dots, \text{ dan } (M \geq M_k)] \\ = \min(M \geq M_i) \end{aligned} \quad (2.13)$$

Perolehan nilai ordinat $d'(A_i) = \min V(S_i \geq S_k)$ untuk $k = 1, 2, \dots, n; k \neq i$. Dari nilai ordinat yang telah diperoleh, kita dapat menentukan nilai vektor bobot yang didefinisikan dengan Persamaan 2.14.

$$W' = (d'(A_1), d'(A_2), \dots, d'(A_n))^T \quad (2.14)$$

Langkah terakhir dari metode F-AHP adalah normalisasi nilai vektor atau nilai prioritas kriteria yang telah diperoleh dengan Persamaan 2.15

$$d(A_n) = \frac{d'(A_n)}{\sum_{i=1}^n d'(A_n)} \quad (2.15)$$

2.4 Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)

Metode *Technique for Order Preference by Similarity to Ideal Solution* atau dikenal dengan TOPSIS adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yoon dan Hwang pada tahun 1981 (Yoon, & Hwang, 1995). TOPSIS memiliki konsep yang mana alternatif yang terpilih merupakan alternatif terbaik yang memiliki jarak terpendek dari solusi ideal positif dan jarak terjauh dari solusi ideal negatif.

Semakin banyaknya faktor yang harus dipertimbangkan dalam proses pengambilan keputusan, maka semakin relatif sulit juga untuk mengambil keputusan terhadap suatu permasalahan. Apalagi jika upaya pengambilan keputusan dari suatu permasalahan tertentu, selain mempertimbangkan berbagai faktor/kriteria yang beragam, juga melibatkan beberapa orang pengambil keputusan. Hal ini dikenal juga dengan permasalahan *Multiple Criteria Decision Making* (MCDM).

Tahapan-tahapan dalam pengerjaan metode TOPSIS adalah sebagai berikut.

1. Menggambarkan alternatif (m) dan kriteria (n) ke dalam sebuah matriks atau matriks keputusan, yang mana X_{ij} adalah pengukuran pilihan dari alternatif ke- i dan kriteria ke- j .
2. Membuat matriks R yaitu matriks keputusan ternormalisasi. Setiap normalisasi dari nilai r_{ij} dapat dilakukan dengan perhitungan menggunakan Persamaan 2.16.

$$r_{ij} = \frac{X_{ij}}{\sqrt{\sum_{i=1}^m X_{ij}^2}} \quad (2.16)$$

3. Membuat pembobotan pada matriks yang telah dinormalisasi Setelah dinormalisasi, setiap kolom pada matriks R dikalikan dengan bobot (W_j) untuk menghasilkan matriks pada Persamaan 2.17.

$$D = \begin{bmatrix} W_1 r_{11} & W_1 r_{12} & W_n r_{1n} \\ W_2 r_{21} & \dots & \dots \\ W_j r_{m1} & W_j r_{m2} & W_j r_{mm} \end{bmatrix} \quad (2.17)$$

4. Menentukan nilai solusi ideal positif dan solusi ideal negatif. Solusi ideal dinotasikan A^+ , sedangkan solusi ideal negatif dinotasikan A^- . Persamaan untuk menentukan solusi ideal dapat dilihat pada Persamaan 2.18 dan 2.19.

$$A^+ = \left\{ \left(\max_{i=1,2,3,\dots,m} V_{ij} \mid j \in J \right), \left(\min_{i=1,2,3,\dots,m} V_{ij} \mid j \in J' \right) \right\} = V_1^+, V_1^+, \dots, V_n^+ \quad (2.18)$$

$$A^- = \left\{ \left(\max_{i=1,2,3,\dots,m} V_{ij} \mid j \in J \right), \left(\min_{i=1,2,3,\dots,m} V_{ij} \mid j \in J' \right) \right\} = V_1^-, V_1^-, \dots, V_n^- \quad (2.19)$$

5. Menghitung *separation measure*. *Separation measure* ini merupakan pengukuran jarak dari suatu alternatif ke solusi ideal positif dan solusi ideal negatif. Perhitungan solusi ideal positif dapat dilihat pada Persamaan 2.20.

$$D_i^+ = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^+)^2} \text{ dengan } i = 1, 2, 3, \dots, m \quad (2.20)$$

Perhitungan solusi ideal negatif dapat dilihat pada Persamaan 2.21.

$$D_i^- = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^-)^2} \text{ dengan } i = 1, 2, 3, \dots, m \quad (2.21)$$

6. Menghitung nilai *preference* untuk setiap alternatif. Untuk menentukan ranking tiap-tiap alternatif yang ada maka perlu dihitung terlebih dahulu nilai preferensi dari tiap alternatif. Perhitungan nilai preferensi dapat dilihat melalui Persamaan 2.22.

$$C_i = \frac{D_i^-}{D_i^+ + D_i^-} \text{ dengan } 0 < C_i < 1 \text{ dan } i = 1, 2, 3, \dots, m \quad (2.22)$$

7. Setelah didapat nilai C_i , maka alternatif dapat diranking berdasarkan urutan C_i . Dari hasil perankingan ini dapat dilihat alternatif terbaik yaitu alternatif yang memiliki jarak terpendek dari solusi ideal dan berjarak terjauh dari solusi ideal negatif.

2.5 Korelasi Spearman (Spearman Corelation)

Pengujian Korelasi Spearman adalah uji statistik yang ditujukan untuk mengetahui hubungan yang dimiliki antara dua atau lebih variabel berskala ordinal dalam penelitian. Formula r_s untuk korelasi Spearman dapat dilihat pada Persamaan 2.23 (Sarwono, 2013).

$$r_s = 1 - \frac{6 \sum_{i=1}^N d_i^2}{N^3 - N} \quad (2.23)$$

r_s = ukuran sampel

N = Banyaknya Observasi

d_i = perbedaan antara kedua peringkat

Nilai r_s dapat bervariasi dari -1 sampai +1. Nilai r_s menunjukkan hubungan yang kuat antara dua variabel tersebut dan nilai r_s yang mendekati 0

mengindikasikan lemahnya hubungan antara dua variabel tersebut. tanda positif dan negatif memberikan informasi mengenai arah hubungan antara dua variabel tersebut. Jika bernilai positif maka kedua variabel memiliki hubungan yang searah. Jika bernilai negatif artinya korelasi antara kedua variabel tersebut bersifat berlawanan (Sarwono, 2013). Hasil dari koefisien korelasi akan diinterpretasikan ke dalam Tabel 2.5.

Tabel 2.5 Koefisien korelasi D.A. de Vaus

| Koefisien | Kekuatan Hubungan |
|-----------|-----------------------------|
| 0,00 | Tidak ada hubungan |
| 0,01-0,09 | Hubungan kurang berarti |
| 0,10-0,29 | Hubungan lemah |
| 0,30-0,49 | Hubungan moderat |
| 0,50-0,69 | Hubungan Kuat |
| 0,70-0,89 | Hubungan Sangat Kuat |
| >0,90 | Hubungan mendekati sempurna |

BAB 3 METODOLOGI

3.1 Studi Kepustakaan

Dalam melakukan penelitian penulis menggunakan metode studi literatur untuk mendapatkan bekal teori acuan yang berhubungan dengan analisis dan implementasi metode *Fuzzy Analytical Hierarchy Process* Dan *Technique For Order Preference By Similarity To Ideal Solution* untuk rekomendasi rumah makan berdasarkan rumah makan yang pernah dikunjungi, di antaranya *Analytic Hierarchy Process*, *Fuzzy Analytical Hierarchy Process*, *Technique for Order Preference by Similarity to Ideal Solution* serta uji korelasi Spearman.

3.2 Tipe Penelitian

Tipe penelitian pada penelitian ini menggunakan tipe non-implementatif dengan pendekatan deskriptif. Pendekatan deskriptif menjelaskan karakteristik objek penelitian dari fenomena/situasi tertentu yang sedang diteliti berdasarkan hasil analisis terhadap data yang diperoleh. Produk/artefak utama yang dihasilkan adalah hasil investigasi.

3.3 Partisipan Penelitian

Penelitian ini melibatkan mahasiswa yang berada di kota Malang yakni 1 mahasiswa Politeknik Negeri Malang laki-laki 21 tahun, 2 mahasiswa Universitas Brawijaya laki-laki 22 tahun dan 24 tahun. Alasan pemilihan partisipan pemilihan ini berdasarkan keterkaitan dengan kebiasaan mahasiswa yang terkadang mencoba berbagai tempat makan.

3.4 Lokasi Penelitian

Lokasi Penelitian adalah di wilayah kota Malang. Hal ini dikarenakan untuk mempermudah dalam pengumpulan data baik rumah makan maupun partisipan penelitian. Penelitian ini tidak membutuhkan tempat khusus dalam pengerjaannya dapat dikerjakan di mana saja di wilayah Malang.

3.5 Pengumpulan Data

Pengumpulan data selama 1 bulan pada tanggal 15 Juni 2017 sampai 10 Juli 2017. Pengumpulan data dilakukan dengan mengamati masing masing kriteria yang dibutuhkan dalam penelitian ini. Kriteria-kriteria yang dibutuhkan adalah jumlah menu, rata-rata harga menu, rating, jarak dan lama waktu buka dari rumah makan. Pengumpulan data di sini dibagi 2 bagian yakni data rumah makan dan data partisipan penelitian yakni mahasiswa yang berada di kota Malang. Pengumpulan data rumah makan yakni jumlah menu, rata-rata harga menu, lokasi dan lama buka dikumpulkan dengan mengobservasi dari aplikasi mobile GOJEK pada menu GoFood sebanyak 30 rumah makan. Hal ini dikarenakan data dari aplikasi ini sudah relevan dengan kondisi rumah makan. Rating Rumah Makan didapat dari rating Google Map Hal ini dikarenakan di aplikasi Gojek tidak tersedia

fitur rating rumah makan. Jarak rumah Makan dapat ditentukan dengan menghitung jarak antara masing-masing partisipan dengan rumah makan melalui sistem yang dibangun.

Pengumpulan data partisipan penelitian adalah lokasi, perbandingan kriteria-kriteria dari penelitian serta penentuan rating masing-masing rumah makan yang pernah dikunjungi partisipan. Lokasi digunakan untuk menentukan jarak rumah makan dengan partisipan penelitian. Perbandingan Kriteria di sini adalah membandingkan kriteria-kriteria dari penelitian sesuai kebutuhan pelanggan. Dan penentuan rating rumah makan yang pernah dikunjungi partisipan digunakan untuk pengujian terhadap hasil yang diperoleh dari sistem.

3.6 Perancangan Sistem

Perancangan sistem dilakukan untuk mengetahui mekanisme proses rekomendasi rumah makan menggunakan metode *Fuzzy Analytical Hierarchy Process* (FAHP) dan *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS) dari data rumah makan serta perbandingan kriteria rumah makan yang pernah dikunjungi oleh pelanggan.

3.7 Implementasi

Implementasi sistem dilakukan dengan menerapkan metode yang telah dikuasai pada studi literatur. Tahapan yang dilakukan dalam implementasi sistem antara lain.

- Implementasi interface menggunakan HTML dan PHP.
- Implementasi metode F-AHP dan TOPSIS ke dalam bahasa pemrograman PHP.

3.8 Pengujian Sistem

Pengujian sistem dilakukan setelah implementasi sistem dilakukan. Pengujian dilakukan untuk mengukur tingkat akurasi hasil pemeringkatan rekomendasi rumah makan terhadap hasil pemeringkatan pakar secara langsung masing-masing pelanggan. Pengujian menggunakan metode uji korelasi Spearman untuk mengetahui hubungan kedekatan antara hasil pemeringkatan system dengan hasil pemeringkatan pakar secara langsung.

3.9 Kesimpulan dan Saran

Pada tahap kesimpulan hasil implementasi yang didapat dari pengujian sistem, dan analisis metode yang diterapkan. Adapun saran, yang ditujukan untuk mengatasi kesalahan yang terjadi dan mempertimbangkan pengembangan penelitian selanjutnya.

BAB 4 PERANCANGAN DAN IMPLEMENTASI

Bab perancangan menjelaskan formulasi permasalahan, siklus algoritme *Fuzzy Analytical Hierarchy Process* (FAHP) dan *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS), siklus penyelesaian rekomendasi rumah makan menggunakan algoritme FAHP-TOPSIS, perhitungan manual, dan perancangan antarmuka.

4.1 Formulasi Permasalahan

Sistem yang akan di bangun dalam penelitian ini adalah sistem rekomendasi makanan menggunakan algoritme *Fuzzy Analytical Hierarchy Process* (FAHP) dan *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS). Data yang digunakan dalam menjalankan sistem ini adalah data hasil kuesioner yang berisi perbandingan antara kriteria dalam merekomendasikan rumah makan yang diinginkan serta data rumah makan di kota Malang yang telah dikunjungi.

Metode FAHP-TOPSIS dijalankan secara sekuensial yang mana pertama kali dilakukan adalah menentukan permasalahan yang akan diselesaikan dan kemudian membaginya menjadi beberapa elemen-elemen kecil dan saling berhubungan. Elemen-elemen ini akan disusun membentuk sebuah hierarki dengan urutan susunan tujuan keputusan, kriteria dan alternatif-alternatif. Pada hierarki ini dibagi menjadi tiga tingkatan sesuai dengan tingkat permasalahan yang dihadapi. Dari hierarki tersebut peneliti akan membuat kuesioner untuk membandingkan antar kriteria dalam untuk menentukan kriteria-kriteria yang telah dipilih dalam menentukan rumah makan yang akan dikunjungi. Berikut salah satu data kuesioner perbandingan kriteria yang digunakan oleh pelanggan 1 telah dikunjungi.

Setelah peneliti memasukkan data hasil dari kuesioner para pengguna, langkah selanjutnya adalah memasukkan data rumah makan di kota Malang yang pernah dikunjungi oleh pengunjung. Berikut adalah contoh data rumah makanan yang pernah dikunjungi oleh pelanggan 1 dapat dilihat pada Tabel 4.1.

Tabel 4.1 Data rumah makan di kota Malang

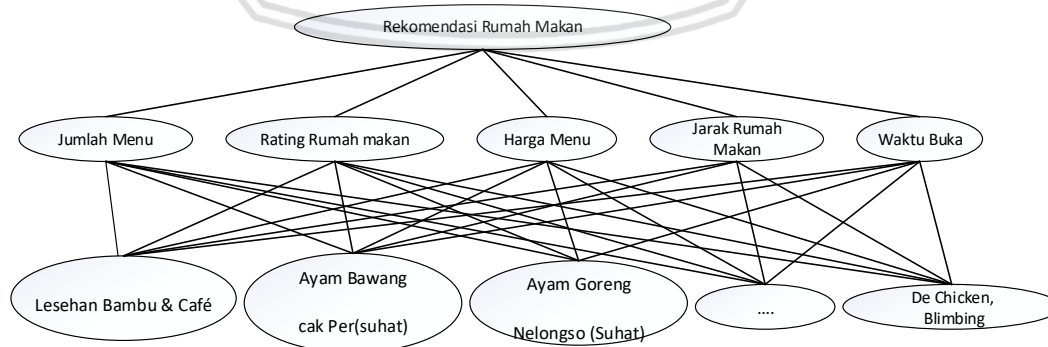
| Alternatif | C1 | C2 | C3 | C4 | C5 |
|------------|---------|-----|------------|-----------|--------|
| A2 | 37 Menu | 4.1 | Rp. 10.351 | 0,5974 Km | 13 Jam |
| A3 | 21 Menu | 4 | Rp. 17.143 | 1,7669 Km | 24 Jam |
| A5 | 23 Menu | 4.2 | Rp. 28.478 | 3,2431 Km | 4 Jam |
| A8 | 11 Menu | 4.4 | Rp. 12.182 | 1,3679 Km | 24 Jam |
| A10 | 8 Menu | 4.3 | Rp. 14.875 | 3,9344 Km | 14 Jam |
| A12 | 11 Menu | 4.1 | Rp. 14.727 | 0,6081 Km | 12 Jam |
| A13 | 24 Menu | 4.2 | Rp. 17.542 | 0,808 Km | 7 Jam |
| A18 | 20 Menu | 4.3 | Rp. 28.025 | 2,0105 Km | 10 Jam |
| A20 | 16 Menu | 4.1 | Rp. 16.938 | 4,2078 Km | 8 Jam |
| A22 | 21 Menu | 4.2 | Rp. 18.095 | 2,4972 Km | 10 Jam |

| | | | | | |
|-----|---------|-----|------------|-----------|--------|
| A23 | 12 Menu | 4.2 | Rp. 23.925 | 6,4516 Km | 11 Jam |
| A26 | 31 Menu | 4.4 | Rp. 13.274 | 0,9641 Km | 11 Jam |
| A28 | 18 Menu | 4.2 | Rp. 18.250 | 2,7479 Km | 9 Jam |
| A30 | 6 Menu | 4 | Rp. 20.167 | 3,1133 Km | 11 Jam |

4.2 Siklus algoritme FAHP-TOPSIS

Siklus algoritme FAHP-TOPSIS merupakan urutan penyelesaian masalah menggunakan algoritme FAHP dan TOPSIS. Masalah yang akan diselesaikan akan dipecah dan disusun dalam bentuk hierarki dengan tiga tingkatan. Tingkatan pertama yakni tujuan keputusan. Tingkatan kedua yakni kriteria yang mana pada tingkatan ini elemen-elemen yang dibentuk saling berhubungan dan dapat dibandingkan pada tingkatan yang sama. Tingkatan Ketiga atau yang terakhir adalah alternatif. Pada tingkatan ini merupakan pilihan dari penyelesaian masalah.

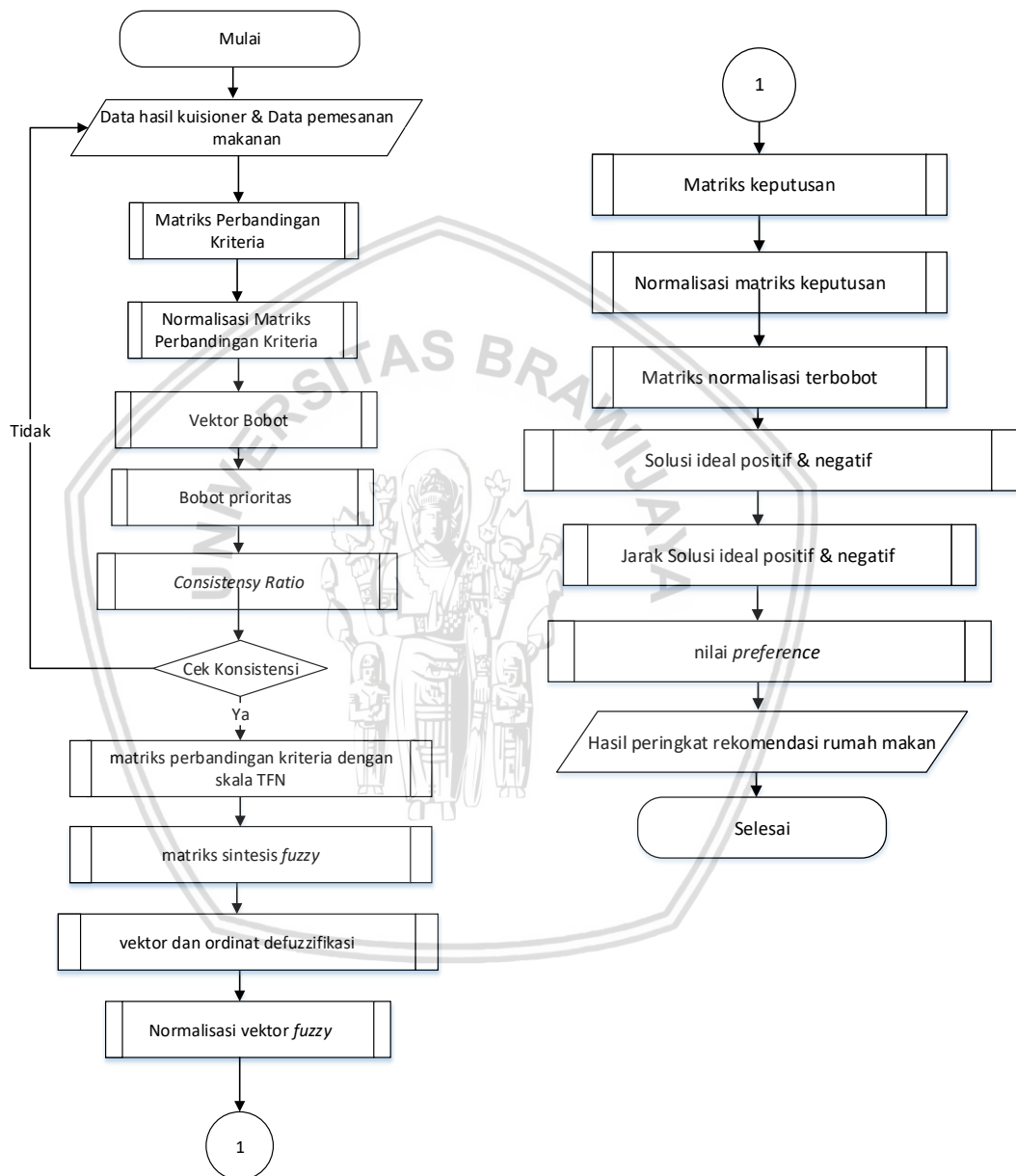
Tujuan Keputusan dari penelitian ini adalah rekomendasi rumah makan. Dalam merekomendasikan rumah makan membutuhkan kriteria-kriteria yang yang digunakan antara lain jumlah menu, rating rumah makan, harga menu, jarak rumah makan dengan alamat pembeli dan waktu buka. jumlah menu dengan rentang 1 sampai dengan 5 menu, 6 sampai dengan 10 menu, 11 sampai dengan 15 menu, 16 sampai dengan 20 menu dan di atas 20 menu. Rating rumah makan meliputi nilai dari menu yang telah dipesan oleh pengguna lain dengan nilai antara 1 sampai 5. Harga menu meliputi rentang 0 sampai dengan 10000, di atas 10000 sampai dengan 20000, di atas 20000 sampai dengan 30000 dan di atas 40000. Jarak rumah makan dengan alamat pembeli dengan rentang 0 sampai dengan 1 km, di atas 1 sampai dengan 2 km, di atas 2 sampai dengan 3 km, di atas 3 sampai dengan 4 km dan di atas 4 km. Kriteria terakhir adalah waktu buka rumah makan dengan rentang 0 sampai 6 jam, di atas 6 jam sampai 12 jam, di atas 12 jam sampai 18 jam dan di atas 18 jam. Alternatif yang akan digunakan dalam penelitian ini adalah data rumah makan di kota Malang.



Gambar 4.1 Hierarki permasalahan

Setelah membuat hierarki permasalahan, peneliti membuat kuesioner perbandingan antar kriteria dalam hierarki tersebut. Kuesioner ini akan diisi oleh pengguna dikarenakan masing-masing pengguna memiliki urutan kebutuhan

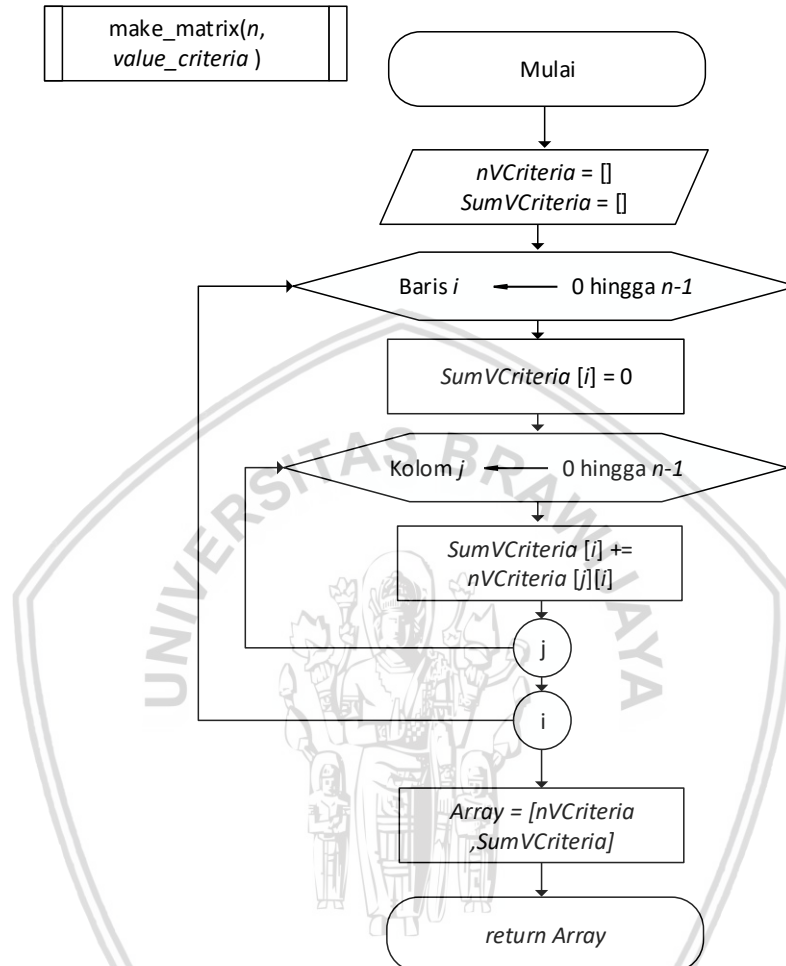
kriteria dalam menentukan hasil rekomendasinya. Peneliti memilih salah satu hasil kuesioner untuk menjalankan proses FAHP-TOPSIS. Metode FAHP digunakan untuk menentukan bobot dari kriteria yang telah dipilih peneliti. Hasil bobot kriteria tersebut akan digunakan dalam proses TOPSIS untuk menentukan hasil rekomendasi yang akan ditampilkan sesuai jumlah data alternatif yang telah dimasukkan oleh peneliti. Berikut adalah alur siklus FAHP-TOPSIS.



Gambar 4.2 Flowchart siklus FAHP-TOPSIS

4.2.1 Matriks Perbandingan Kriteria

Pada Tahap pembuatan matriks perbandingan kriteria dengan langkah-langkah sesuai Gambar 4.3.

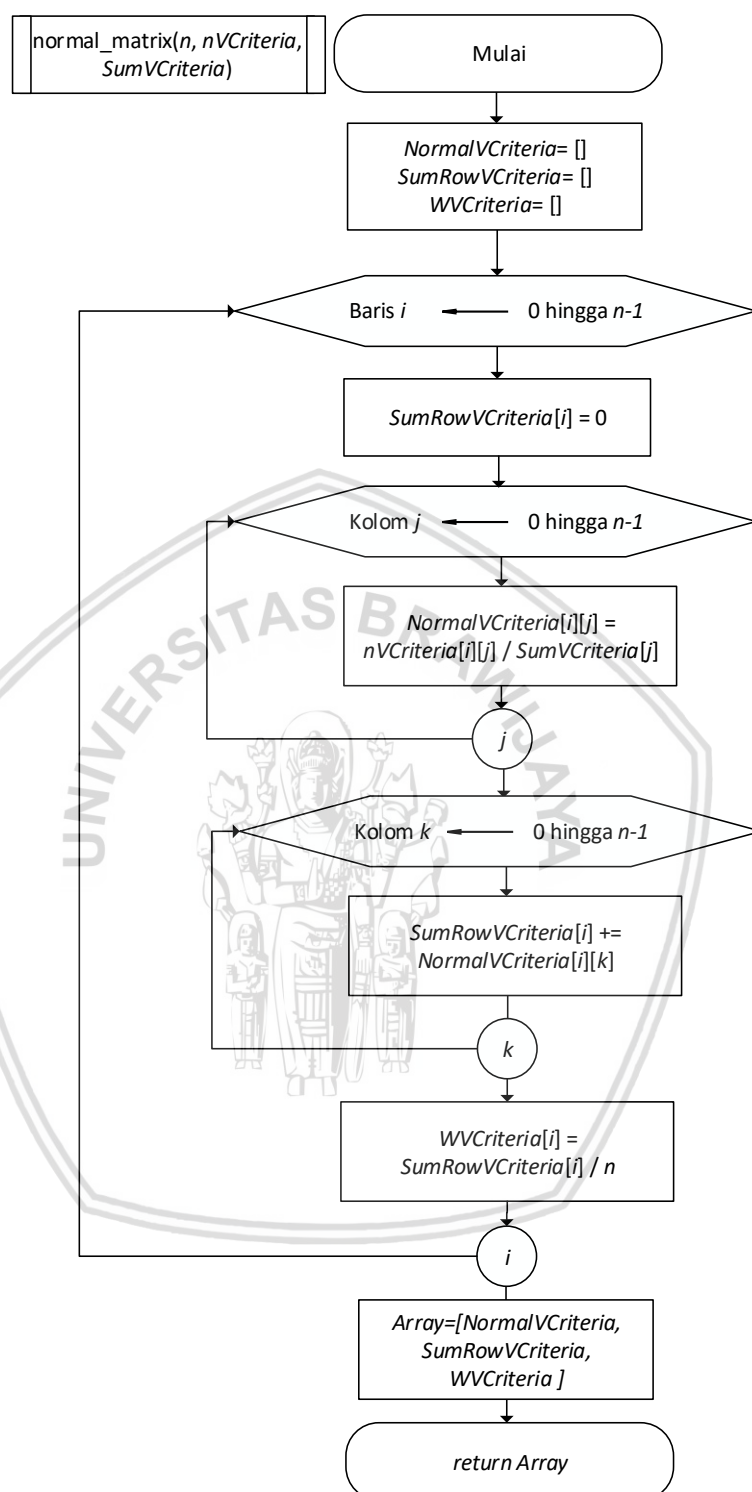


Gambar 4.3 Flowchart matriks perbandingan kriteria

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria dan $value_criteria$ yakni array berisi nilai perbandingan kriteria. Kemudian inisialisasi array $nVCriteria$ dan $SumVCriteria$. Array $nVCriteria$ diberi nilai dengan nilai $value_criteria$ dengan $index$ row dan col . Kemudian lakukan perulangan baris i sebanyak $n-1$ dan kolom j sebanyak $n-1$ untuk memberi nilai array $SumVCriteria$. Nilai $SumVCriteria$ didapat dari penjumlahan $nVCriteria$ sesuai dengan kolom j . Selanjutnya array $nVCriteria$ dan $SumVCriteria$ di simpan ke dalam variabel $Array$ dan keluarkan dengan nilai kembalian berupa variabel $Array$.

4.2.2 Normalisasi Matriks Perbandingan Kriteria

Pada Tahap ini matriks perbandingan kriteria yang telah dibuat akan dinormalisasi sesuai dengan langkah-langkah sesuai Gambar 4.4.



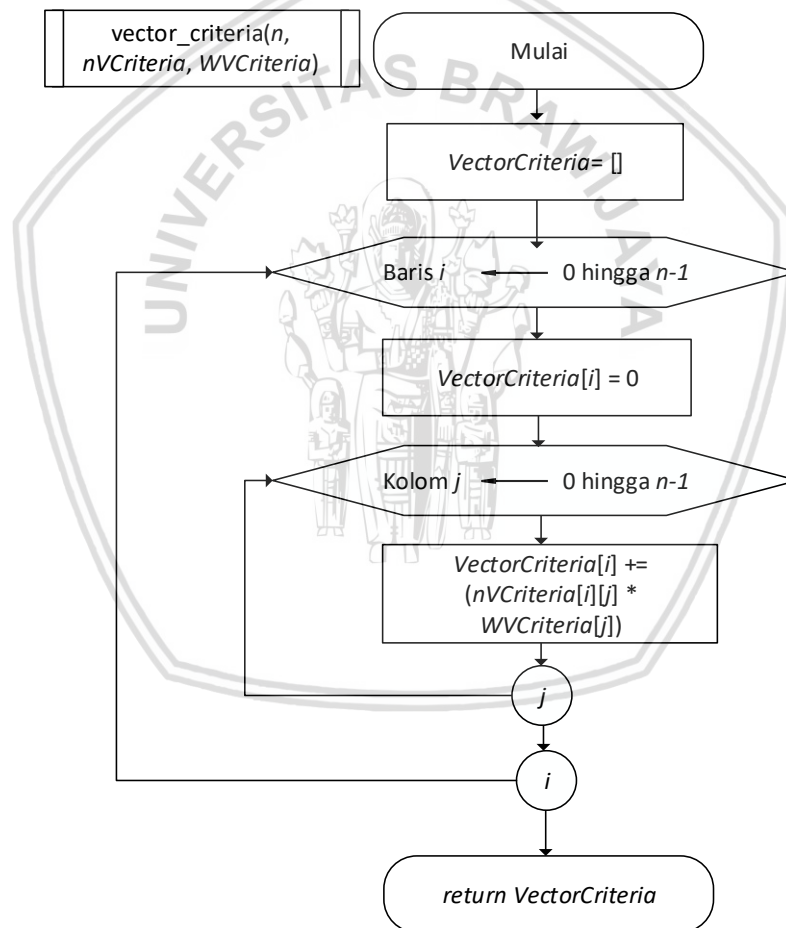
Gambar 4.4 Flowchart normalisasi matriks perbandingan kriteria

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, $nVCriteria$ yakni array berisi nilai perbandingan kriteria dan $SumVCriteria$ yakni array berisi jumlah $nVCriteria$ sesuai kolomnya. Kemudian inisialisasi array

NormalVCriteria, *SumRowVCriteria* dan *WVCriteria*. Lakukan perulangan baris *i* sebanyak *n-1* dan kolom *j* sebanyak *n-1* untuk pemberian nilai array *NormalVCriteria* dengan membagi antara nilai *nVCriteria* dengan *SumVCriteria*. Kemudian lakukan perulangan kolom *k* sebanyak *n* untuk memberi nilai array *SumRowVCriteria* dengan menjumlahkan semua *NormalVCriteria* pada baris *i*. Selanjutnya pengisian nilai array *WVCriteria* dari pembagian *SumRowVCriteria* dengan *n*. Terakhir array *NormalVCriteria*, *SumRowVCriteria* dan *WVCriteria* di simpan ke dalam variabel Array dan keluarkan dengan nilai kembalian berupa variabel Array.

4.2.3 Vektor Bobot

Setelah normalisais matriks, akan dilakukan perhitungan vektor bobot sesuai dengan langkah-langkah sesuai Gambar 4.5.



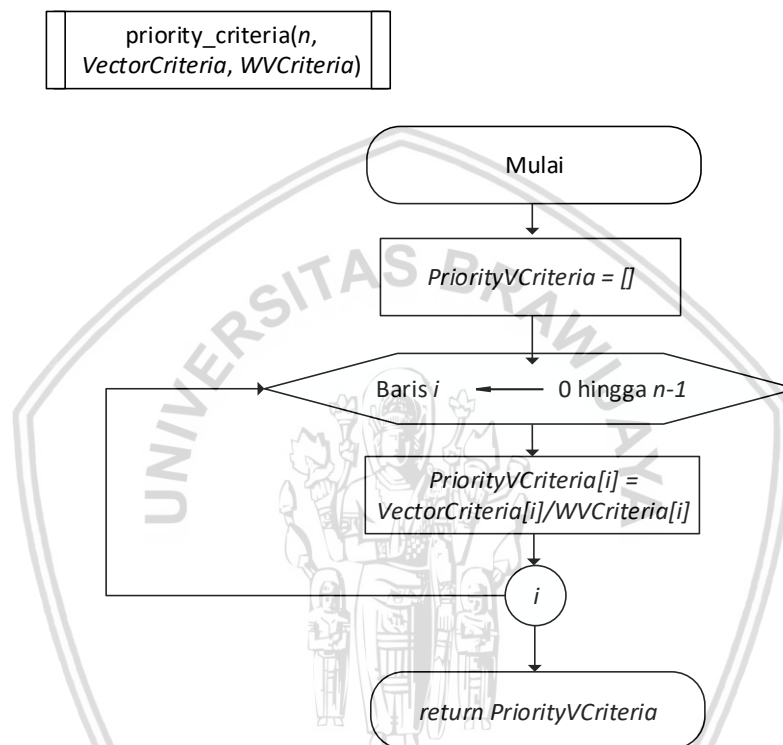
Gambar 4.5 Flowchart perhitungan bobot prioritas

Proses diawali dengan memasukkan parameter berupa *n* yakni banyak kriteria, *nVCriteria* yakni array berisi nilai perbandingan kriteria dan *WVCriteria* yakni array berisi bobot awal kriteria. Kemudian inisialisasi array *VectorCriteria*. Lakukan perulangan baris *i* sebanyak *n-1* dan berikan nilai awal array

VectorCriteria 0. Kemudian didalam perulangan baris i , lakukan perulanngan kolom j sebanyak $-1n$ untuk pemberian nilai *array* *VectorCriteria* dengan menjumlah perkalian antara nilai *nVCriteria* dengan *WVCriteria*. Keluarkan dengan nilai kembalian berupa variabel *WVCriteria*.

4.2.4 Nilai Prioritas

Setelah vektor bobot, akan dilakukan perhitungan bobot priotitas sesuai dengan langkah-langkah sesuai Gambar 4.6.

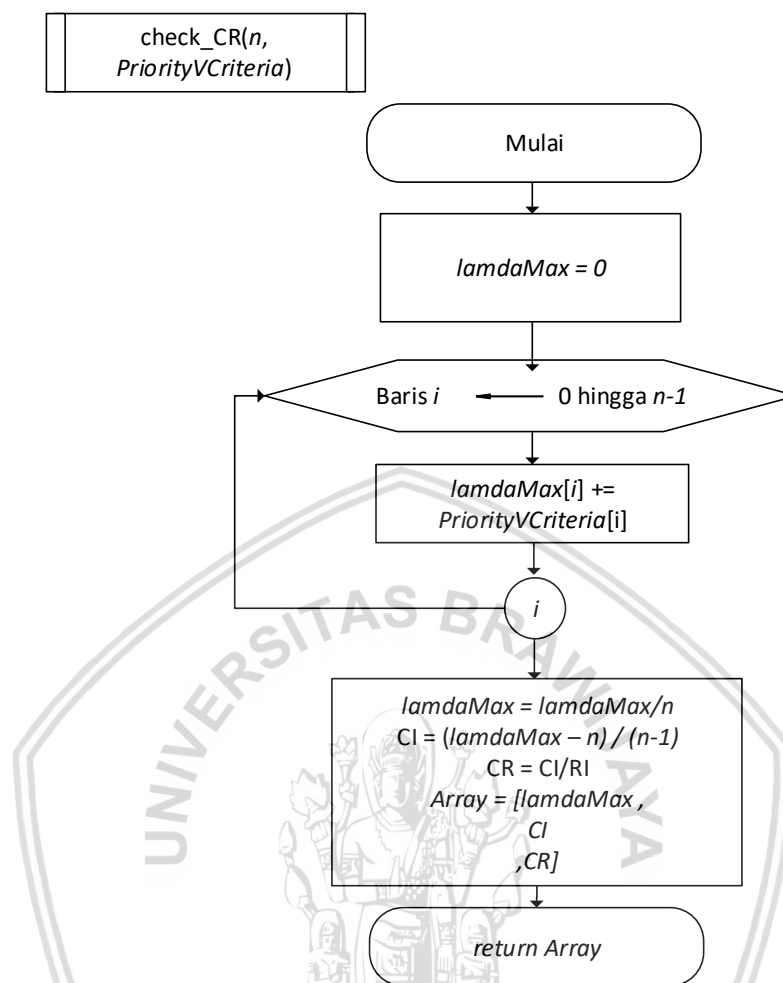


Gambar 4.6 Flowchart perhitungan bobot prioritas

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, *VectorCriteria* yakni *array* berisi nilai vektor bobot dan *WVCriteria* yakni *array* berisi bobot awal kriteria. Kemudian inisialisasi *array* *PriorityVCriteria*. Lakukan perulangan baris i sebanyak $n-1$ untuk pemberian nilai *array* *PriorityVCriteria* dengan membagi antara nilai *VectorCriteria* dengan *WVCriteria*. Keluarkan dengan nilai kembalian berupa variabel *PriorityVCriteria*.

4.2.5 Consistency Ratio

Setelah didapat Bobot Prioritas yang di peroleh akan digunakan untuk menghitung nilai *consistency ratio*. Langkah-langkah sesuai Gambar 4.7.



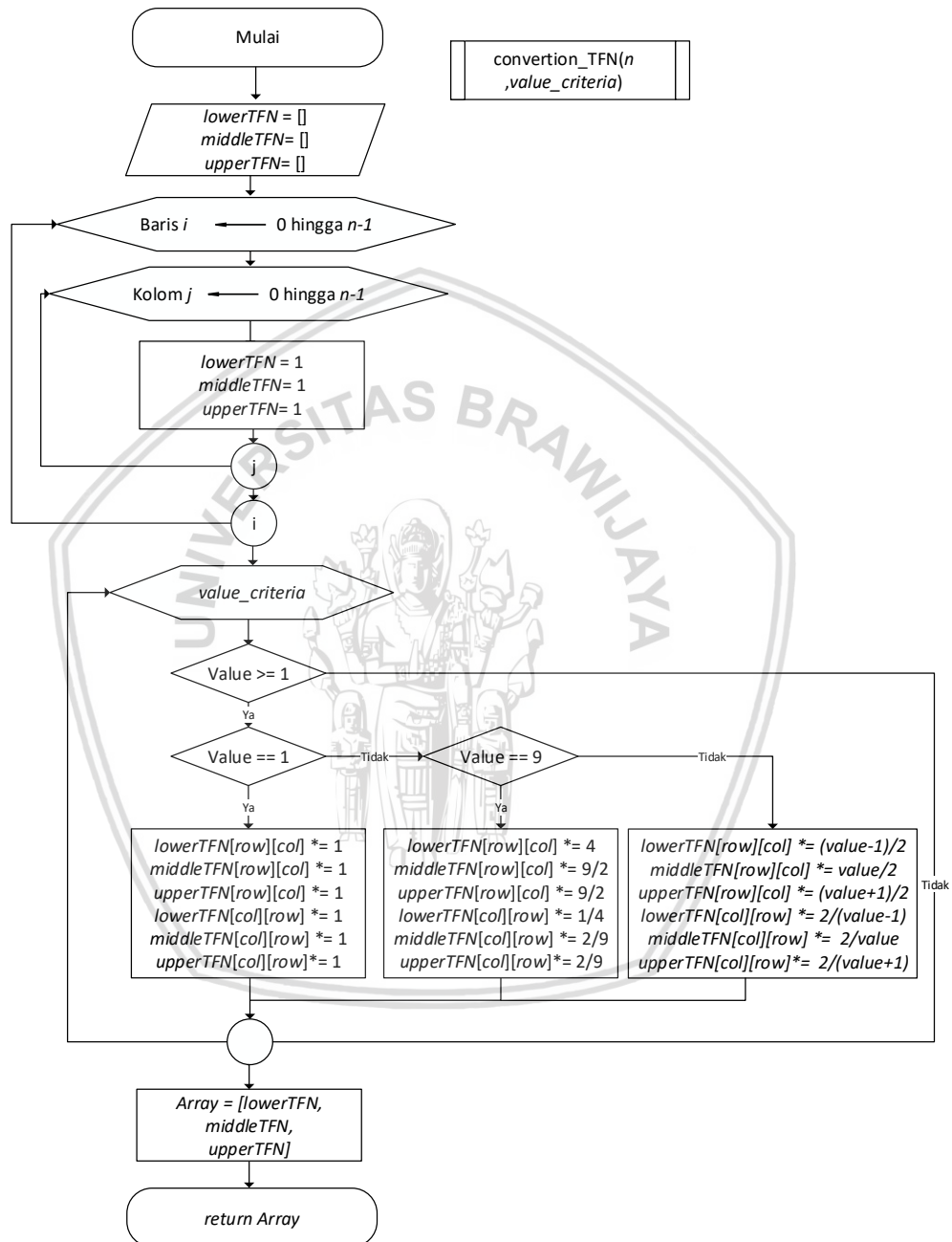
Gambar 4.7 Flowchart consistency ratio

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, *PriorityVCriteria* yakni array berisi bobot prioritas. Kemudian inisialisasi variabel *lamdaMax* dengan nilai 0. Lakukan perulangan baris i sebanyak $n-1$ untuk memberikan nilai *lamdaMax* dengan jumlah *PriorityVCriteria*. Kemudian membagi nilai *lamdaMax* dengan n . Setelah itu mencari nilai variabel CI dengan membagi *lamdaMax* dikurang n dengan n dikurang 1. Inisialisasi variabel CR dengan membagi CI dengan RI yakni 1,12 dikarenakan jumlah kriteria yang digunakan sebanyak 5 sesuai Tabel 2.3. Terakhir *lamdaMax*, CI dan CR di simpan ke dalam variabel *Array* dan keluarkan dengan nilai kembalian berupa variabel *Array*.

Menurut Saaty, jika nilai $CR \leq 10\%$ atau $CR \leq 0,1$ maka matriks perbandingan yang digunakan konsisten. Hal ini dapat diartikan bahwa semua kriteria telah dikelompokkan secara *homogeny* dan relasi antar kriteria saling membenarkan. Peneliti dapat melanjutkan proses selanjutnya. Namun jika hasil $CR > 10\%$ atau $CR > 0,1$ maka harus dipilih lagi matriks perbandingan kriteria dari sumber yang lain dan peneliti mengulangi proses dari normalisasi dengan matriks yang baru.

4.2.6 Matriks Perbandingan Kriteria Dengan Skala TFN

Matriks perbandingan yang telah lolos cek konsistensi akan diubah menjadi matriks fuzzifikasi dengan skala TFN seperti pada Tabel 2.8. Langkah-langkah sesuai Gambar 4.8.



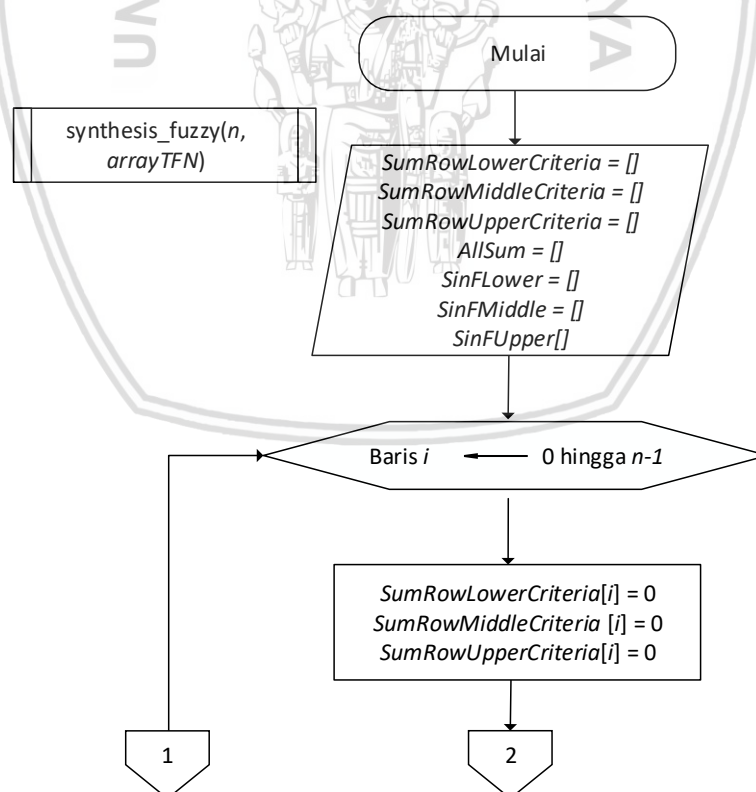
Gambar 4.8 Flowchart proses matriks skala TFN

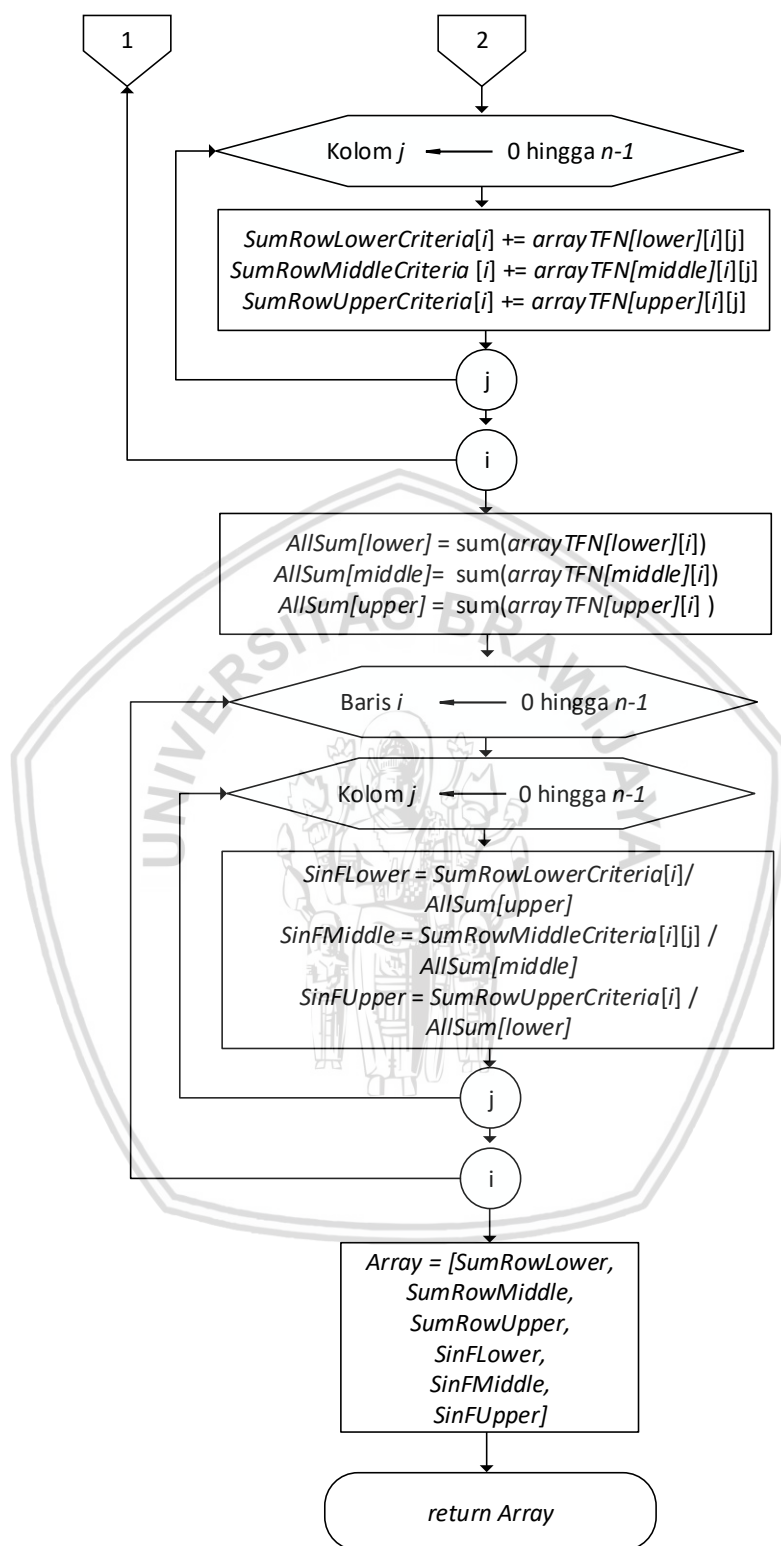
Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, $value_criteria$ yakni array berisi kriteria perbandingan dari database. Kemudian inisialisasi variabel array $lowerTFN$, array $middleTFN$ dan array $upperTFN$. Lakukan perulangan baris i sebanyak $n-1$ dan perulangan kolom

sebanyak $n-1$ untuk memberikan nilai *lowerTFN*, *middleTFN* dan *upperTFN* dengan nilai 1. Kemudian lakukan perulangan sebanyak data pada *array value_criteria*. Di dalam perulangan tadi, terdapat seleksi jika nilai *value* dari *array value_criteria* lebih dari 1 akan di proses selain itu akan diabaikan. Setelah lolos seleksi awal, akan dilakukan seleksi kembali. Jika nilai *value* sama dengan 1 maka nilai *lowerTFN*, *middleTFN* dan *upperTFN* dengan *index (row, col)* dan *(col, row)* akan bernilai 1. Selain itu jika nilai *value* sama dengan 9 maka nilai *lowerTFN*, *middleTFN* dan *upperTFN* dengan *index (row, col)* bernilai 4, 9/2, 9/2 serta nilai *lowerTFN*, *middleTFN* dan *upperTFN* dengan *index (col, row)* bernilai 1/4, 2/9, 2/9. Sedangkan jika tidak sesuai dengan kedua seleksi maka nilai *lowerTFN*, *middleTFN* dan *upperTFN* dengan *index (row, col)* bernilai $(value-1)/2$, $value/2$, $(value+1)/2$ serta nilai *lowerTFN*, *middleTFN* dan *upperTFN* dengan *index (col, row)* bernilai $2/(value-1)$, $2/value$, $2/(value+1)$. Terakhir *lowerTFN*, *middleTFN* dan *upperTFN* di simpan ke dalam variabel *Array* dan keluarkan dengan nilai kembalian berupa variabel *Array*.

4.2.7 Matriks Sintesis Fuzzy

Matriks fuzzifikasi yang telah dibuat akan digunakan sebagai masukan untuk pembuatan matriks sintetis fuzzy. Berikut Langkah-langkah sesuai Gambar 4.9.





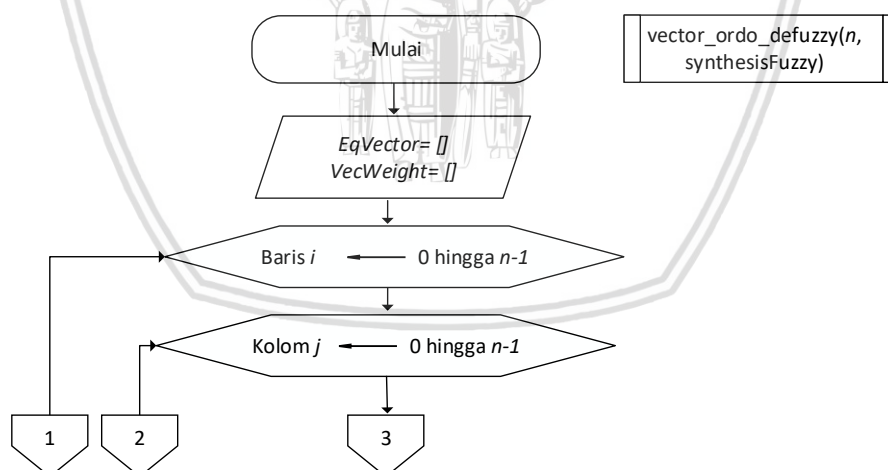
Gambar 4.9 Flowchart proses matriks sintesis fuzzy

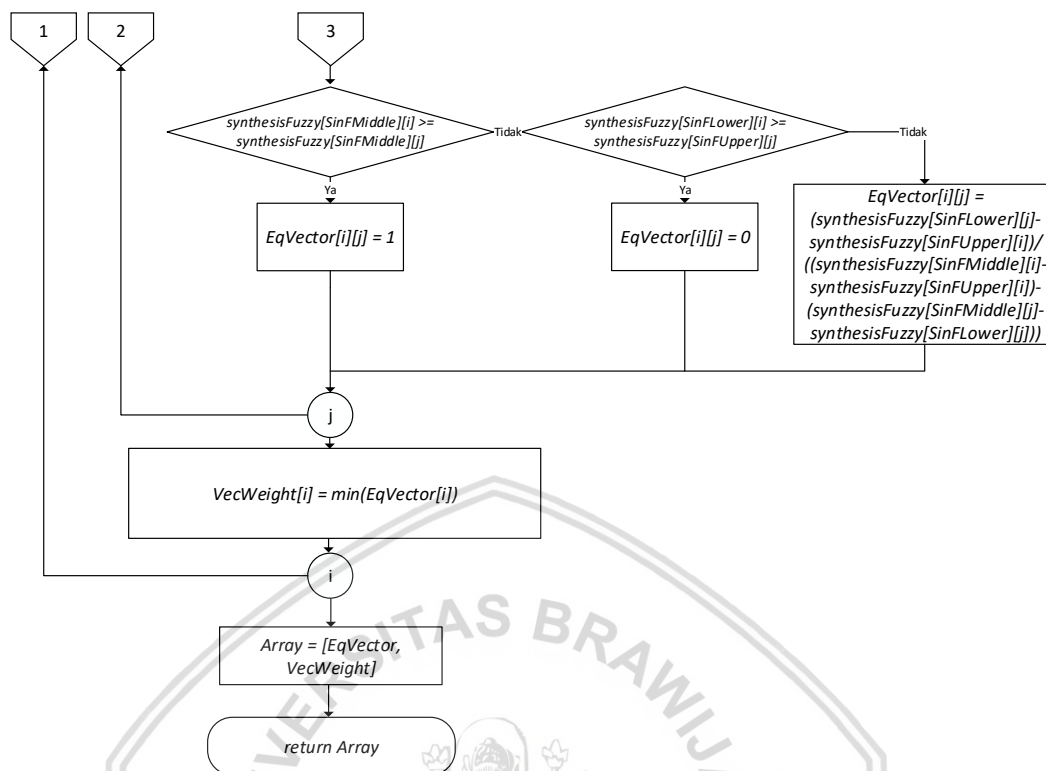
Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, $arrayTFN$ yakni $array$ berisi matriks yang telah dikonversi dengan skala

TFN. Kemudian inisialisasi variabel array *SumRowLowerCriteria*, *SumRowMiddleCriteria*, *SumRowUpperCriteria*, *AllSum*, *SinFLower*, *SinFMiddle*, dan *SinFUpper*. Lakukan perulangan baris *i* sebanyak *n-1* untuk memberikan nilai *SumRowLowerCriteria*, *SumRowMiddleCriteria* dan *SumRowUpperCriteria* dengan nilai 0. Kemudian di lakukan perulangan kolom *j* sebanyak *n-1* untuk memberikan nilai *SumRowLowerCriteria* dengan penjumlahan *arrayTFN[lower]*, *SumRowMiddleCriteria* dengan penjumlahan *arrayTFN[middle]*, *SumRowUpperCriteria* dengan penjumlahan *arrayTFN[upper]* sesuai dengan kriteria masing-masing. Kemudian keluar dari kedua perulangan sebelumnya memberikan nilai *AllSum[lower]* jumlah semua nilai *arrayTFN[lower]*, *AllSum[middle]* jumlah semua nilai *arrayTFN[middle]* dan *AllSum[upper]* jumlah semua nilai *arrayTFN[upper]*. Lakukan kembali perulangan baris *i* sebanyak *n* dan perulangan kolom *j* sebanyak *n* untuk memberikan nilai *SinFLower* dengan membagi nilai *SumRowLowerCriteria* dengan *AllSum[upper]*, *SinFMiddle* dengan membagi nilai *SumRowMiddleCriteria* dengan *AllSum[middle]* dan *SinFUpper* dengan membagi nilai *SumRowUpperCriteria* dengan *AllSum[lower]* sesuai dengan kriteria masing-masing. Terakhir *SumRowLower*, *SumRowMiddle*, *SumRowUpper*, *SinFLower*, *SinFMiddle* dan *SinFUpper* di simpan ke dalam variabel Array dan keluarkan dengan nilai kembalian berupa variabel Array.

4.2.8 Vektor dan Ordinat Defuzzifikasi

Matriks sintesis fuzzy akan digunakan untuk mencari nilai vektor dan defuzzifikasi. Berikut Langkah-langkah sesuai Gambar 4.10.



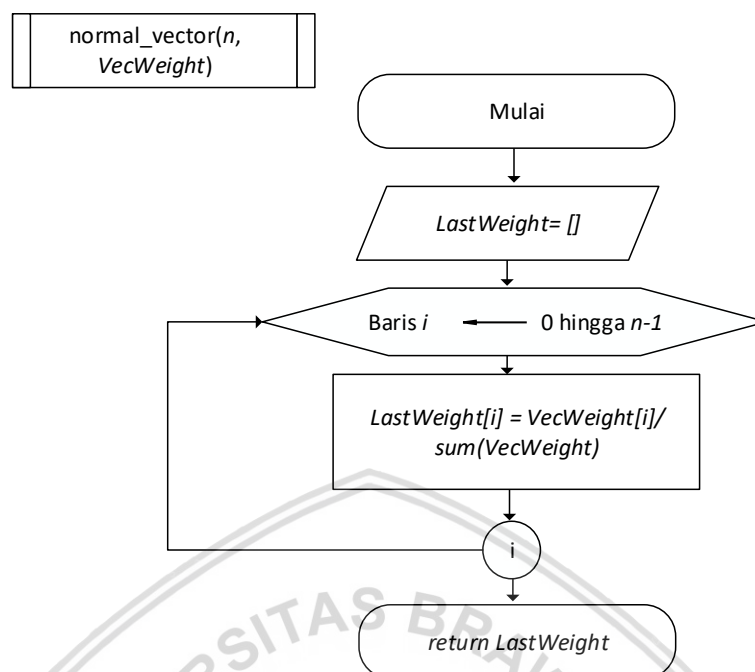


Gambar 4.10 Flowchart Proses perhitungan vektor dan ordinat defuzzifikasi

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, *synthesisFuzzy* yakni array berisi matriks sintesis fuzzy. Kemudian inisialisasi variabel array *EqVector* dan array *VecWeight*. Lakukan perulangan baris i sebanyak $n-1$ dan perulangan kolom j sebanyak $n-1$. Di dalam perulangan tadi, terdapat seleksi jika nilai *synthesisFuzzy[SinFMiddle][i]* lebih besar sama dengan *synthesisFuzzy[SinFMiddle][j]* maka nilai *EqVector[i][j]* sama dengan 1. Kemudian jika nilai *synthesisFuzzy[SinFLower][i]* lebih besar sama dengan *synthesisFuzzy[SinFUpper][j]* maka nilai *EqVector[i][j]* sama dengan 0. Selain dari ke dua seleksi itu maka nilai $EqVector[i][j] = \frac{(synthesisFuzzy[SinFLower][j] - synthesisFuzzy[SinFUpper][i])}{((synthesisFuzzy[SinFMiddle][i] - synthesisFuzzy[SinFUpper][i]) - (synthesisFuzzy[SinFMiddle][j] - synthesisFuzzy[SinFLower][j]))}$. Setelah itu diluar perulangan kolom j memberikan nilai *VecWeight* dengan nilai minimal *EqVector* sesuai dengan kriteria masing-masing. Terakhir *EqVector* dan *VecWeight* di simpan ke dalam variabel *Array* dan keluarkan dengan nilai kembalian berupa variabel *Array*.

4.2.9 Normalisasi Vektor Fuzzy

Matriks fuzzy akan digunakan sebagai masukkan untuk proses normalisasi nilai vektor fuzzy atau bobot akhir kriteria. Berikut Langkah-langkah sesuai Gambar 4.11.

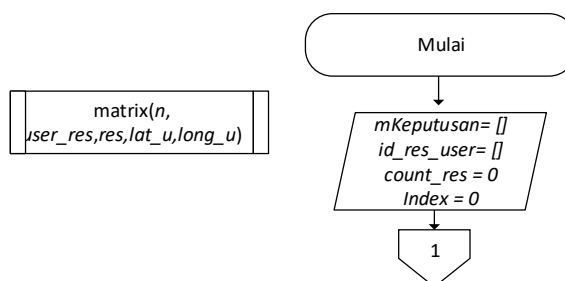


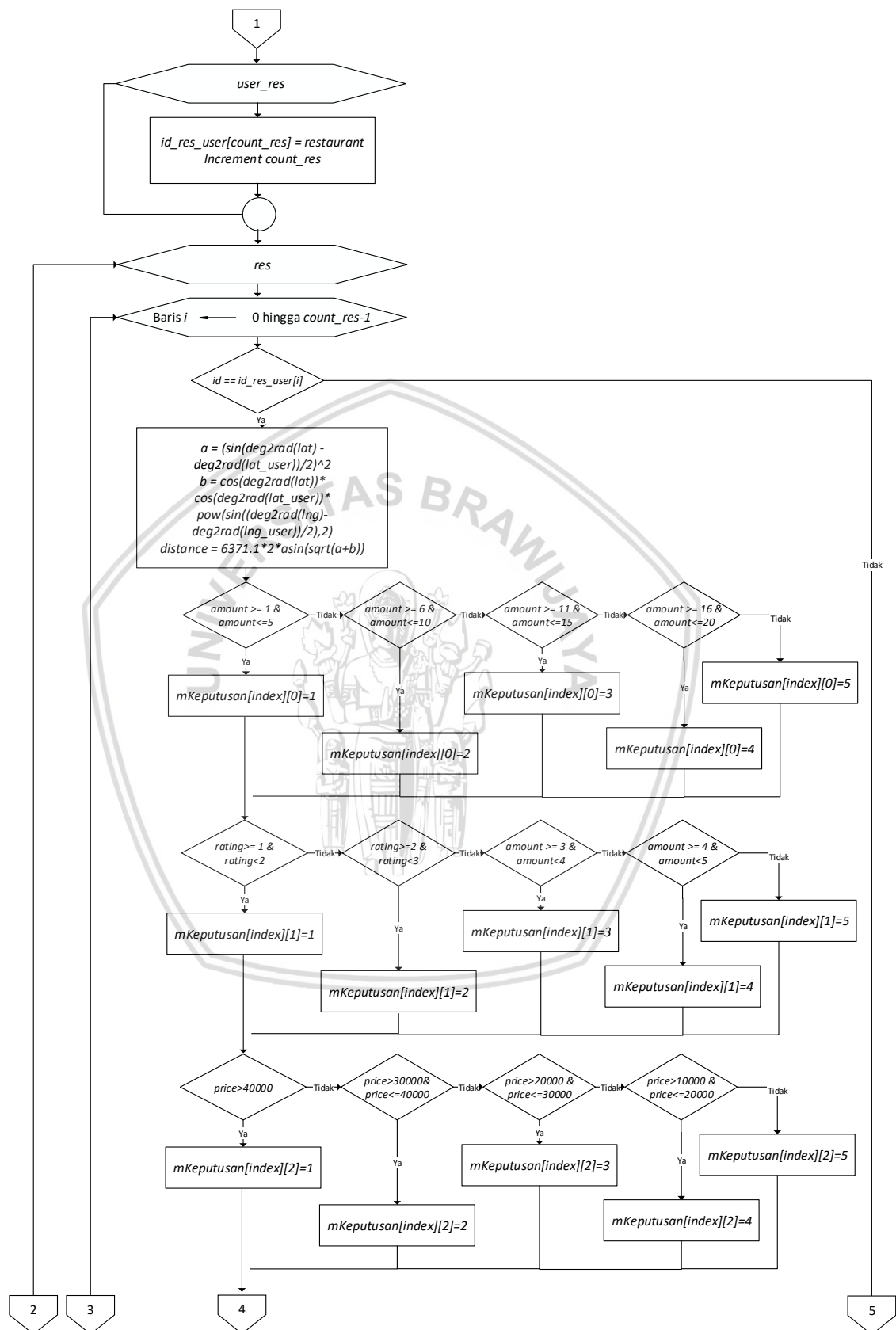
Gambar 4.11 Flowchart normalisasi vektor fuzzy

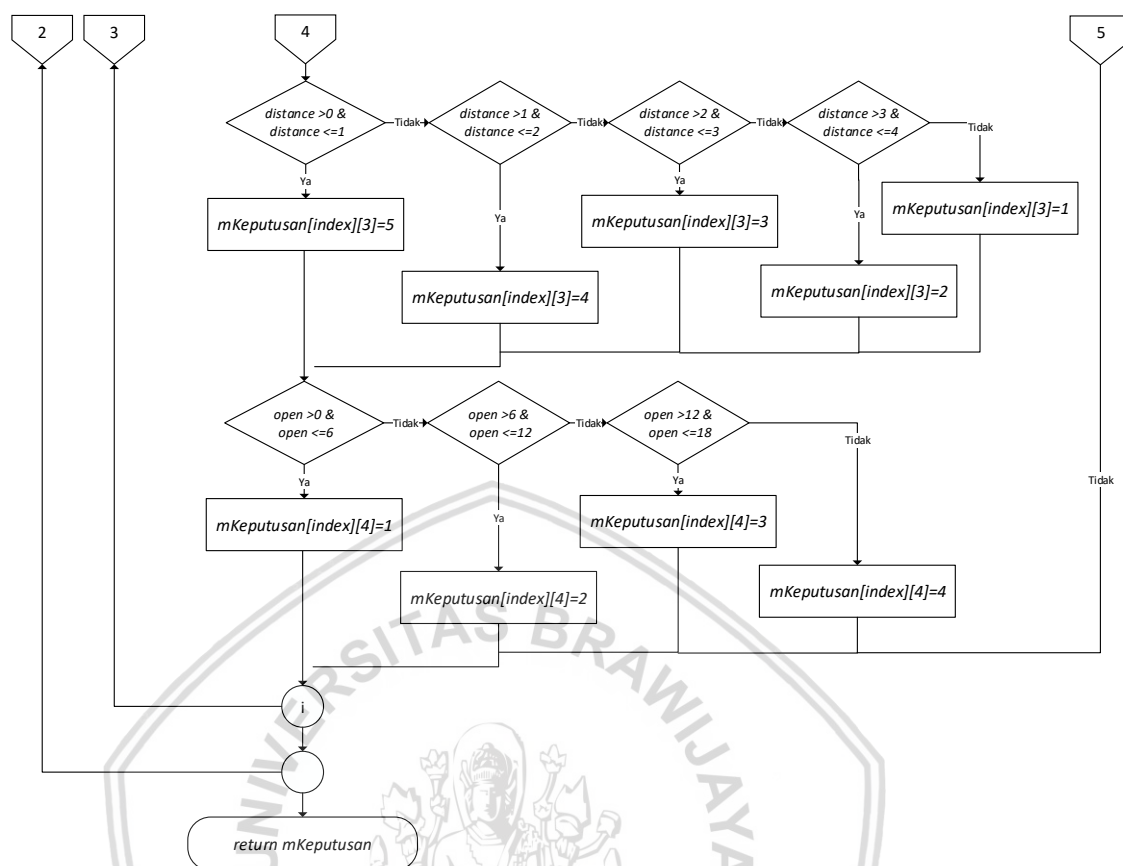
Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, $VecWeight$ yakni array berisi matriks vektor dan ordinat defuzzifikasi. Kemudian inisialisasi variabel array $LastWeight$. Lakukan perulangan baris i sebanyak $n-1$ untuk memberikan nilai $LastWeight[i]$ dengan membagi $VecWeight[i]$ dengan jumlah semua $VecWeight$. keluarkan dengan nilai kembalian berupa $LastWeight$.

4.2.10 Matriks keputusan

Matriks keputusan dibuat dari proses perubahan nilai dari masing-masing kriteria rumah makan ke dalam skala ordinal. Berikut Langkah-langkah sesuai Gambar 4.12.







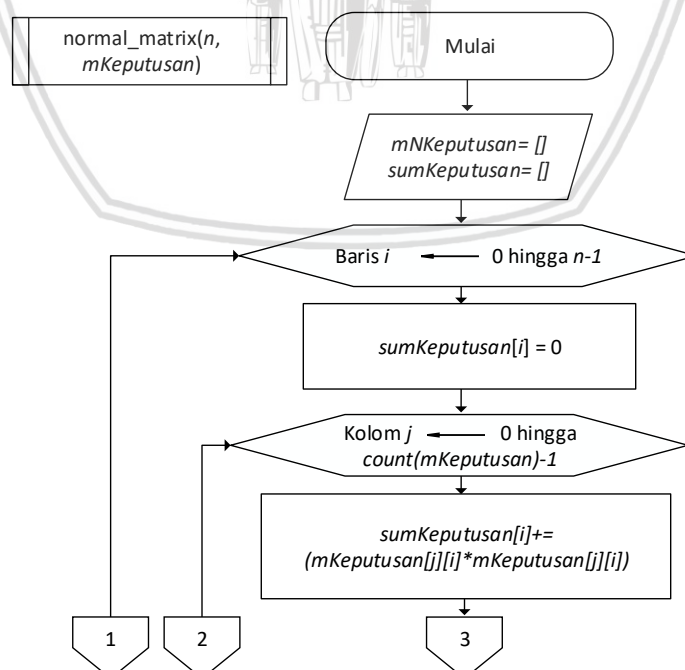
Gambar 4.12 Flowchart normalisasi vektor fuzzy

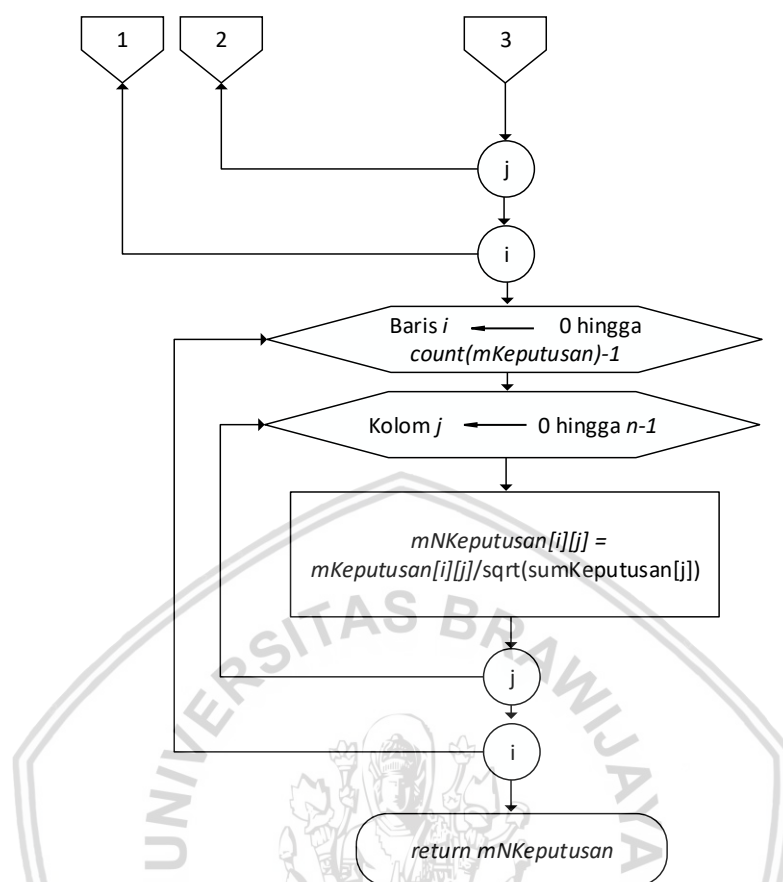
Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, $user_res$ yakni array rumah makan yang pernah dikunjungi oleh pelanggan, res yakni array data input kriteria rumah makan serta lat_user dan lng_user adalah data *latitude* dan *longitude* pelanggan berasal. Kemudian inisialisasi variabel array $mKeputusan$, array id_res_user serta $count_res$ dan $index$ dengan nilai awal 0. Lakukan perulangan sebanyak data $user_res$ untuk mengisi nilai id_res_user dengan $id_restaurant$ dan menambahkan nilai $count_res$ dengan 1. Lakukan Perulangan sebanyak data res dan perulangan baris i sebanyak $count_res-1$. Di dalam perulangan ini terjadi seleksi jika id sama dengan id_res_user maka akan dilakukan statemen di dalamnya. Jika tidak akan diabaikan. Setelah itu pemberian nilai variabel a dengan $(\sin(deg2rad(lat) - deg2rad(lat_user))/2)^2$, variabel b dengan nilai $\cos(deg2rad(lat)) * \cos(deg2rad(lat_user)) * \text{pow}(\sin((deg2rad(lng) - deg2rad(lng_user))/2), 2)$ dan variabel $distance$ dengan nilai $6371,1 * 2 * \text{asin}(\text{sqrt}(a+b))$. Setelah itu terdapat seleksi jika $amount \geq 1$ dan $amount \leq 5$ maka nilai $mKeputusan[index][0] = 1$. Sedangkan jika $amount \geq 6$ dan $amount \leq 10$ maka nilai $mKeputusan[index][0] = 2$. Sedangkan jika $amount \geq 10$ dan $amount \leq 15$ maka nilai $mKeputusan[index][0] = 3$. Sedangkan jika $amount \geq 15$ dan $amount \leq 20$ maka nilai $mKeputusan[index][0] = 4$. Dan Jika tidak memenuhi semuanya maka nilai $mKeputusan[index][0] = 5$. Setelah itu terdapat seleksi kembali jika $rating \geq 1$ dan

$rating < 2$ maka nilai $mKeputusan[index][1] = 1$. Sedangkan jika $rating \geq 2$ dan $rating < 3$ maka nilai $mKeputusan[index][1] = 2$. Sedangkan jika $rating \geq 3$ dan $rating < 4$ maka nilai $mKeputusan[index][1] = 3$. Sedangkan jika $rating \geq 4$ dan $rating < 5$ maka nilai $mKeputusan[index][1] = 4$. Dan Jika tidak memenuhi semuanya maka nilai $mKeputusan[index][1] = 5$. Setelah itu terdapat seleksi kembali jika $price > 40000$ maka nilai $mKeputusan[index][2] = 1$. Sedangkan jika $price > 30000$ dan $price \leq 40000$ maka nilai $mKeputusan[index][2] = 2$. Sedangkan jika $price > 20000$ dan $price \leq 30000$ maka nilai $mKeputusan[index][2] = 3$. Sedangkan jika $price > 10000$ dan $price \leq 20000$ maka nilai $mKeputusan[index][2] = 4$. Dan Jika tidak memenuhi semuanya maka nilai $mKeputusan[index][2] = 5$. Setelah itu terdapat seleksi kembali jika $distance > 0$ dan $distance \leq 1$ maka nilai $mKeputusan[index][3] = 1$. Sedangkan jika $distance > 1$ dan $distance \leq 2$ maka nilai $mKeputusan[index][3] = 2$. Sedangkan jika $distance > 2$ dan $distance \leq 3$ maka nilai $mKeputusan[index][3] = 3$. Sedangkan jika $distance > 3$ dan $distance \leq 4$ maka nilai $mKeputusan[index][3] = 4$. Dan Jika tidak memenuhi semuanya maka nilai $mKeputusan[index][3] = 5$. Dan seleksi terakhir jika $open > 0$ dan $open \leq 6$ maka nilai $mKeputusan[index][4] = 1$. Sedangkan jika $open > 6$ dan $open \leq 12$ maka nilai $mKeputusan[index][4] = 2$. Sedangkan jika $open > 12$ dan $open \leq 18$ maka nilai $mKeputusan[index][4] = 3$. Dan Jika tidak memenuhi semuanya maka nilai $mKeputusan[index][4] = 4$. Diluar perulangan akan dikeluarkan dengan nilai kembalian berupa variabel $mKeputusan$.

4.2.11 Normalisasi matriks keputusan

Matriks keputusan digunakan sebagai masukan untuk proses normalisasi matriks keputusan. Berikut Langkah-langkah sesuai Gambar 4.13.



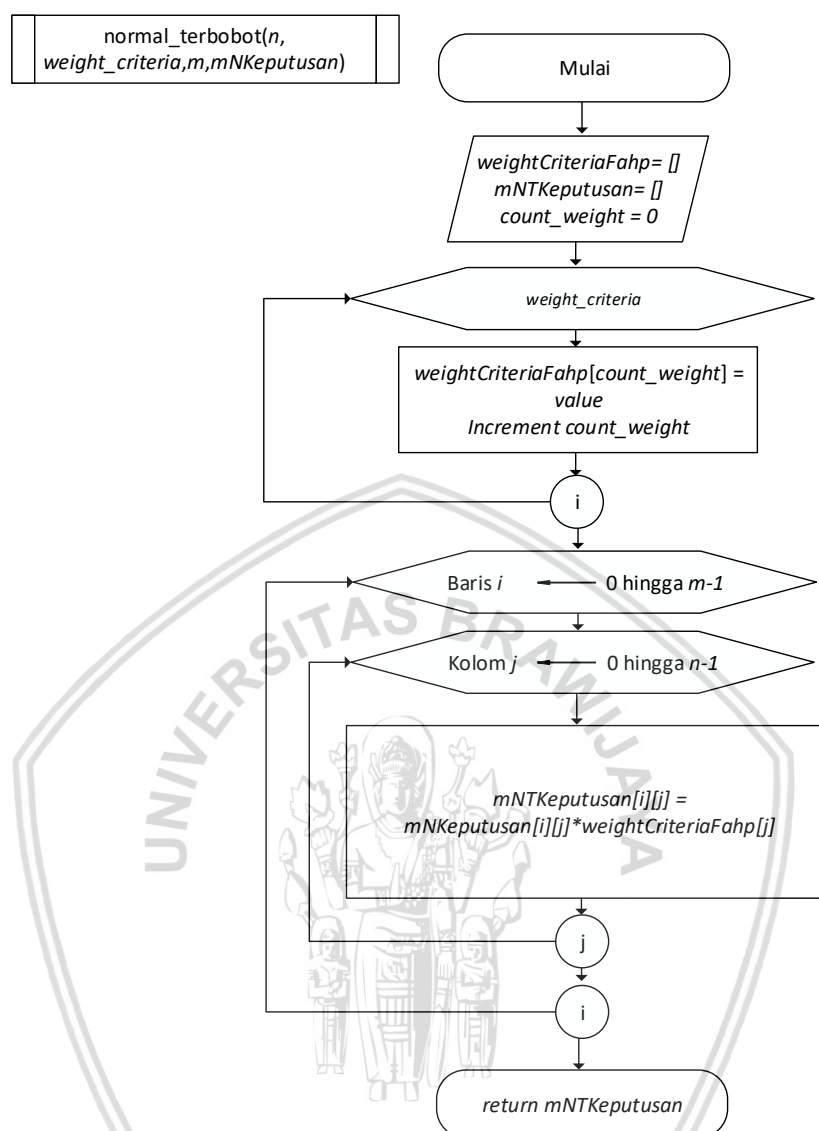


Gambar 4.13 Flowchart matriks keputusan ternormalisasi

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, $mKeputusan$ yakni array berisi matriks keputusan. Kemudian inisialisasi variabel array $mNKeputusan$ dan array $sumKeputusan$. Lakukan perulangan baris i sebanyak $n-1$ dan perulangan j sebanyak $count(mKeputusan)-1$. Di dalam perulangan baris i , terdapat pemberian nilai terhadap $sumKeputusan$ dengan nilai awal 0. Pada perulangan kolom j terdapat pemberian nilai terhadap $sumKeputusan$ dengan menjumlahkan perkalian $mKeputusan[j][i]$ dengan $mKeputusan[j][i]$. Setelah keluar dari perulangan, Lakukan perulangan kembali. Lakukan perulangan baris i sebanyak $count(mKeputusan)-1$ dan perulangan kolom j sebanyak $n-1$. Didalam perulangan terdapat pemberian nilai terhadap $mNKeputusan$ dengan membagi $mKeputusan$ dengan $sqrt(sumKeputusan)$. Keluarkan dengan nilai kembalian berupa variabel $mNKeputusan$.

4.2.12 Matriks Normalisasi Terbobot

Matriks normalisasi keputusan digunakan sebagai masukan untuk proses pembobotan matriks normalisasi keputusan. Berikut Langkah-langkah sesuai Gambar 4.14.

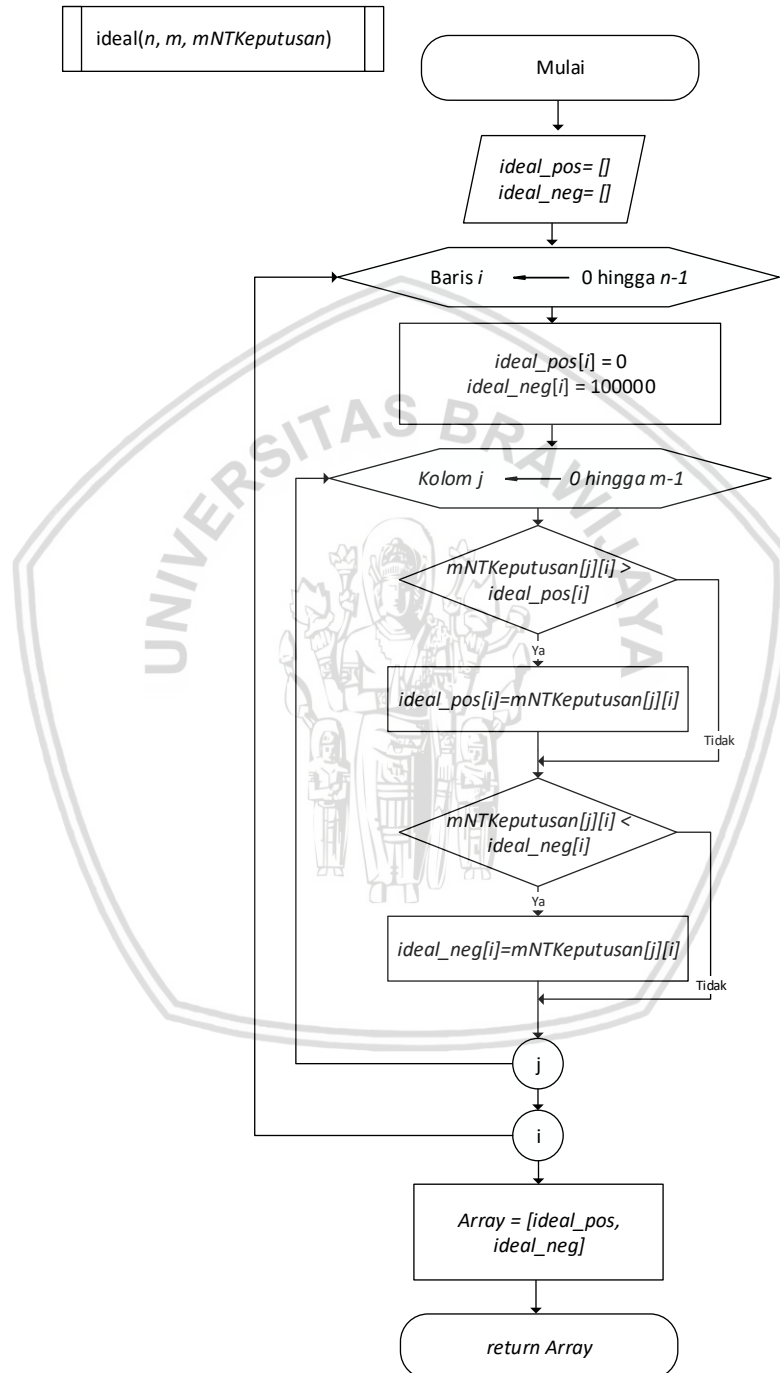


Gambar 4.14 Flowchart matriks normalisasi terbobot

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, $weight_criteria$ yakni array berisi matriks bobot kriteria dari proses FAHP, m banyaknya baris matriks keputusan dan $mNKeputusan$ yakni array normalisasi matriks keputusan. Kemudian inisialisasi variabel array $weightCriteriaFahp$ dan array $mNTKeputusan$ serta variabel $count_weight$ dengan nilai 0. Lakukan perulangan sebanyak data $weight_criteria$ untuk memberikan nilai terhadap $weightCriteriaFahp$ dengan $value$ serta penambahan 1 terhadap $count_weight$. Lakukan perulangan baris i sebanyak $m-1$ dan perulangan j sebanyak $n-1$. Di dalam perulangan baris i dan kolom j terdapat pemberian nilai terhadap $mNTKeputusan$ dengan menjumlahkan perkalian $mNKeputusan$ dengan $weightCriteriaFahp$. Keluarkan dengan nilai kembalian berupa variabel $mNTKeputusan$.

4.2.13 Solusi Ideal Positif dan Solusi Ideal Negatif

Matriks normalisasi terbobot digunakan sebagai masukan proses pencarian solusi ideal positif dan solusi ideal negatif. Berikut Langkah-langkah sesuai Gambar 4.15.



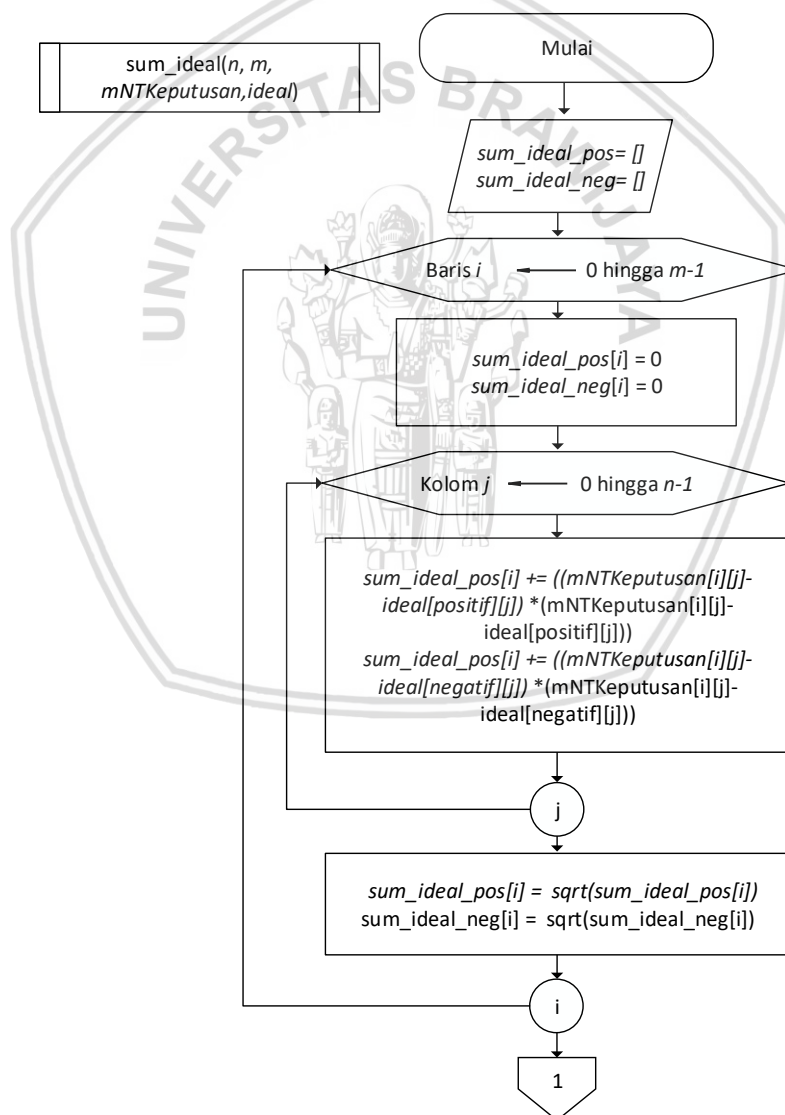
Gambar 4.15 Flowchart solusi ideal positif dan solusi ideal negatif

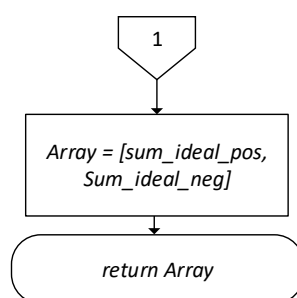
Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, m banyaknya baris matriks keputusan dan $mNTKeputusan$ yakni array

matriks normalisasi terbobot. Kemudian inialisasi variabel *array ideal_pos* dan *array ideal_neg*. Lakukan perulangan baris *i* sebanyak *n-1* dan memberikan nilai *ideal_pos* dengan nilai awal 0 dan *ideal_neg* dengan nilai awal 100000 serta perulangan *j* sebanyak *m-1*. Di dalam perulangan kolom *j* terdapat seleksi jika $mNTKeputusan > ideal_pos$ maka $ideal_pos = mNTKeputusan$, selain itu akan diabaikan. Seleksi selanjutnya ika $mNTKeputusan < ideal_neg$ maka $ideal_neg = mNTKeputusan$, selain itu akan diabaikan. Terakhir *ideal_pos* dan *ideal_neg* di simpan ke dalam variabel *Array* dan keluarkan dengan nilai kembalian berupa variabel *Array*.

4.2.14 Jarak Solusi Ideal Positif dan Jarak Solusi Ideal Negatif

Matriks ideal positif-negatif akan diproses pencarian jarak solusi ideal positif dan jarak solusi ideal negatif. Berikut Langkah-langkah sesuai Gambar 4.16.



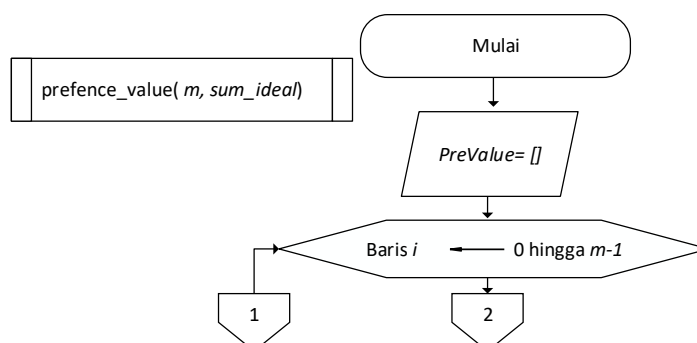


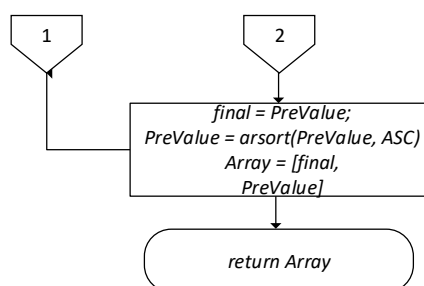
Gambar 4.16 Flowchart jarak ideal positif dan jarak ideal negatif

Proses diawali dengan memasukkan parameter berupa n yakni banyak kriteria, m banyaknya baris matriks keputusan, $mNTKeputusan$ yakni array matriks normalisasi terbobot dan array ideal yang berisi matriks solusi ideal positif dan negatif. Kemudian inisialisasi variabel array sum_ideal_pos dan array sum_ideal_neg . Lakukan perulangan baris i sebanyak $m-1$ dan memberikan nilai $ideal_pos$ dengan nilai awal 0 dan $ideal_neg$ dengan nilai awal 0 serta perulangan j sebanyak $n-1$. Di dalam perulangan kolom j terdapat pemberian nilai sum_ideal_pos dengan penjumlahan dari perkalian antara pengurangan $mNTKeputusan$ dan $ideal[positif]$ dengan pengurangan $mNTKeputusan$ dan $ideal[positif]$ dan sum_ideal_neg dengan penjumlahan dari perkalian antara pengurangan $mNTKeputusan$ dan $ideal[negatif]$ dengan pengurangan $mNTKeputusan$ dan $ideal[negatif]$. Keluar dari perulangan j , nilai sum_ideal_pos diperbarui dengan akar dari nilai akhir sum_ideal_pos dan nilai sum_ideal_neg diperbarui dengan akar dari nilai akhir sum_ideal_neg . Terakhir sum_ideal_pos dan sum_ideal_neg di simpan ke dalam variabel Array dan keluarkan dengan nilai kembalian berupa variabel Array.

4.2.15 Nilai Preference

Matriks jarak ideal positif-negatif dari proses sebelumnya akan digunakan sebagai masukan untuk proses pencarian nilai *preference*. Berikut Langkah-langkah sesuai Gambar 4.16.





Gambar 4.17 Flowchart pencarian nilai *preference*

Proses diawali dengan memasukkan parameter berupa m banyaknya baris matriks keputusan dan sum_ideal berisi array dari jarak solusi ideal positif dan negatif. Kemudian inisialisasi variabel array $PreValue$. Lakukan perulangan baris i sebanyak $m-1$ dan memberikan nilai $PreValue$ dengan membagi antara $sum_ideal[negatif]$ dengan penjumlahan $sum_ideal[positif]$ dan $sum_ideal[negatif]$. Setelah keluar dari perulangan i membuat variabel array $final$ berisi nilai dari array $PreValue$ dan array $PreValue$ dilakukan pengurutan dari terbesar ke terkecil. Terakhir $final$ dan $PreValue$ di simpan ke dalam variabel $Array$ dan keluarkan dengan nilai kembalian berupa variabel $Array$.

Urutan alternatif yang dibuat menunjukan digunakan untuk mengurutkan data rumah makan. Dan akan di tampilkan dalam daftar nama rumah makan.

4.3 Siklus Penyelesaian Rekomendasi Rumah Makan Menggunakan Algoritme FAHP-TOPSIS

Siklus algoritme FAHP-TOPSIS telah diuraikan sebelumnya yang selanjutnya akan disederhanakan dalam perhitungan manual agar memudahkan dalam memahami penyelesaian permasalahan rekomendasi rumah makan, sebelum diimplementasikan ke dalam kode program.

4.3.1 Matriks Perbandingan Kriteria

Dalam proses ini merupakan tahap awal dari metode FAHP akan ditentukan nilai perbandingan prioritas kriteria dengan cara membandingkan seluruh kriteria secara berpasangan. Nilai yang diberikan untuk seluruh perbandingan diperoleh dari skala perbandingan 1-9 pada Tabel 2.2. Hasil yang diperoleh dapat dilihat pada Tabel 4.2.

Tabel 4.2 Perbandingan kriteria

| Kriteria | C1 | C2 | C3 | C4 | C5 |
|----------|------|----|--------|-----|--------|
| C1 | 1 | 4 | 0,1667 | 0,2 | 1 |
| C2 | 0,25 | 1 | 0,125 | 0,2 | 0,3333 |
| C3 | 6 | 8 | 1 | 2 | 9 |
| C4 | 5 | 5 | 0,5 | 1 | 1 |
| C5 | 1 | 3 | 0,1111 | 1 | 1 |

Pada Tabel 4.2 C1, C2, C3, C4 dan C5 merupakan kriteria yang telah ditentukan dalam merekomendasikan rumah makan. Nilai masukkan diperoleh dari pengguna dalam membandingkan antara 2 kriteria dalam kuesioner. Misal dalam perbandingan C1 dengan C2 memiliki nilai 0,25 dari 1.4. Hal ini dikarenakan C1 lebih diprioritaskan daripada C2. Hasil akhir dari proses ini adalah nilai perbandingan C2 dengan C1 adalah 4.

4.3.2 Normalisasi Matriks Perbandingan Kriteria

Matriks perbandingan kriteria akan dinormalisasi pada tahapan ini. Normalisasi dilakukan dengan membagi nilai matriks dengan jumlah dari semua nilai matriks pada suatu kolom kriteria.

Setiap nilai *cell* pada Tabel 4.2 akan dinormalisasi menggunakan Persamaan (2.1). Normalisasi nilai matriks pada kolom 1 baris 1 dengan urutan seperti di bawah ini.

$$\text{Normalisasi} = \frac{1}{1+0,25+6+5+1} = \frac{1}{13,25} = 0,0755$$

Demikian seterusnya hingga kolom dan baris terakhir dari matriks. Hasil akhir proses ini berupa dapat dilihat pada Tabel 4.3.

Tabel 4.3 Normalisasi matriks perbandingan kriteria

| Kriteria | C1 | C2 | C3 | C4 | C5 |
|----------|--------|--------|--------|--------|--------|
| C1 | 0,0755 | 0,1905 | 0,0876 | 0,0455 | 0,0811 |
| C2 | 0,0189 | 0,0476 | 0,0657 | 0,0455 | 0,027 |
| C3 | 0,4528 | 0,381 | 0,5255 | 0,4545 | 0,7297 |
| C4 | 0,3774 | 0,2381 | 0,2628 | 0,2273 | 0,0811 |
| C5 | 0,0755 | 0,1429 | 0,0584 | 0,2273 | 0,0811 |

Kemudian dicari nilai bobot kriteria awal dengan membagi jumlah dari nilai matriks normalisasi pada setiap baris kriteria dengan banyaknya kriteria.

Dari Tabel 4.3, untuk memperoleh nilai bobot prioritas menggunakan Persamaan 2.2. Nilai bobot prioritas kriteria 1 diperoleh dengan urutan.

$$\text{Bobot Prioritas} = \frac{0,755+0,1905+0,0876+0,0455+0,0811}{5} = \frac{0,4802}{5} = 0,096$$

Demikian seterusnya hingga kolom dan baris terakhir dari matriks. Bobot prioritas kriteria diperoleh dapat dilihat pada Tabel 4.4.

Tabel 4.4 Bobot kriteria awal

| Kriteria | Bobot Kriteria Awal |
|----------|---------------------|
| C1 | 0,096 |
| C2 | 0,0409 |
| C3 | 0,5087 |
| C4 | 0,2373 |
| C5 | 0,117 |

4.3.3 Vektor Bobot

Hasil dari bobot kriteria awal akan digunakan untuk mencari vektor bobot. Vektor bobot dapat diperoleh dengan penjumlahan dari perkalian antara kriteria perbandingan dengan bobot kriteria awal.

Dari Tabel 4.2 dan Tabel 4.4, untuk memperoleh nilai vektor bobot dapat menggunakan Persamaan 2.3.

$$\text{Vektor bobot} = (1 \times 0,096) + (4 \times 0,0409) + (0,1667 \times 0,5087) + (0,2 \times 0,2373) + (1 \times 0,1170) = 0,509$$

Demikian seterusnya hingga kolom dan baris terakhir dari matriks. Vektor bobot diperoleh dapat dilihat pada Tabel 4.5.

Tabel 4.5 Vektor bobot

| Kriteria | Vektor Jumlah Bobot |
|----------|---------------------|
| C1 | 0,509 |
| C2 | 0,215 |
| C3 | 2,940 |
| C4 | 1,2934 |
| C5 | 0,6297 |

4.3.4 Nilai Prioritas

Hasil dari vektor bobot akan digunakan untuk mencari nilai prioritas. Nilai prioritas didapat dari pembagian vektor bobot dengan bobot kriteria awal.

Dari Tabel 4.4 dan Tabel 4.5, untuk memperoleh nilai prioritas dapat menggunakan dengan Persamaan 2.4.

$$\text{Prioritas} = \frac{0,509}{0,096} = 5,3013$$

Demikian seterusnya hingga kriteria terakhir. Bobot prioritas diperoleh dapat dilihat pada Tabel 4.6.

Tabel 4.6 Nilai prioritas kriteria

| Kriteria | Nilai Prioritas |
|----------|-----------------|
| C1 | 5,3013 |
| C2 | 5.2524 |
| C3 | 5.7793 |
| C4 | 5.4503 |
| C5 | 5.3811 |

4.3.5 Consistency Ratio

Pada tahapan ini *consistency ratio* akan dicari terlebih dahulu nilai Eigen maks (λ_{maks}), *CI* (*Consistency Index*) dan *RI* (*Ratio Index*). *RI* diperoleh dari Tabel 2.7 Nilai *RI*, sehingga nilai *RI* adalah 1,12 dikarenakan banyak kriteria yang dimiliki 5. *CI* diperoleh dengan mengurangi nilai λ_{maks} dengan banyaknya kriteria kemudian dibagi dengan banyak kriteria dikurangi 1. Dan λ_{maks} didapat dari penjumlahan semua nilai prioritas kemudian dibagi dengan jumlah kriteria. Berikut adalah tahapan-tahapan perhitungannya.

1. Mencari nilai Eigen maks dengan Persamaan 2.5.

$$\begin{aligned}\lambda_{maks} &= \frac{5.3013+5.2524+5.7793+5.4503+5.3811}{5} \\ &= 5.4329\end{aligned}$$

2. Mencari nilai *CI* dengan Persamaan 2.6.

$$CI = \frac{5,4329-5}{4} = 0,1082$$

3. Mencari *CR* dengan Persamaan 2.7.

$$CR = \frac{0,1082}{1,12} = 0,0966$$

Menurut Saaty, jika nilai $CR \leq 10\%$ atau $CR \leq 0,1$ maka matriks perbandingan yang digunakan konsisten. Hal ini dapat diartikan bahwa semua kriteria telah dikelompokkan secara homogen dan relasi antar kriteria saling membenarkan. Peneliti dapat melanjutkan proses selanjutnya. Namun jika hasil $CR > 10\%$ atau $CR > 0,1$ maka harus dipilih lagi matriks perbandingan kriteria dari sumber yang lain dan peneliti mengulangi proses dari normalisasi dengan matriks yang baru. Dari perhitungan akhir dapat diketahui jika nilai $CR \leq 10\%$, sehingga proses selanjutnya dapat dilanjutkan.

4.3.6 Matriks Perbandingan Kriteria dengan Skala TFN

Pada tahapan ini matriks perbandingan kriteria dari Tabel 4.2 akan dikonversi dengan menggunakan Skala *Triangular Fuzzy Number* (TFN) pada Tabel 2.8 dari skala AHP. Dari masing-masing data perbandingan kriteria akan menghasilkan 3 data yakni *lower*, *middle* dan *upper*. Hasil konversi dari matriks tersebut dapat dilihat pada Tabel 4.7.

Tabel 4.7 Konversi matriks dengan skala TFN

| Kriteria | C1 | | | C2 | | | C3 | | | C4 | | | C5 | | |
|-----------|-----|-----|--------|-----|-----|-----|--------|--------|--------|--------|-----|-----|-----|--------|-----|
| | l | m | u | l | m | u | l | m | u | l | m | u | l | m | u |
| C1 | 1 | 1 | 1 | 1,5 | 2 | 2,5 | 0,2857 | 0,3333 | 0,4 | 0,3333 | 0,4 | 0,5 | 1 | 1 | 1 |
| C2 | 0,4 | 0,5 | 0,6667 | 1 | 1 | 1 | 0,2222 | 0,25 | 0,2857 | 0,3333 | 0,4 | 0,5 | 0,5 | 0,6667 | 1 |
| C3 | 2,5 | 3 | 3,5 | 3,5 | 4 | 4,5 | 1 | 1 | 1 | 0,5 | 1 | 1,5 | 4 | 4,5 | 4,5 |
| C4 | 2 | 2,5 | 3 | 2 | 2,5 | 3 | 0,6667 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| C5 | 1 | 1 | 1 | 1 | 1,5 | 2 | 0,25 | 0,2222 | 0,2222 | 1 | 1 | 1 | 1 | 1 | 1 |

4.3.7 Matriks Sintesis Fuzzy

Konversi matriks dengan skala TFN atau dikenal dengan matriks fuzzifikasi akan digunakan untuk membuat matriks baru yakni matriks sintesis fuzzy. Matriks ini diperoleh dengan mencari nilai jumlah masing-masing kolom *lower*, *middle* dan *upper* setiap baris dan jumlah dari semua nilai pada kolom *lower*, *middle* dan *upper*. Hasil dari pencarian 2 perhitungan tersebut akan digunakan untuk menghitung nilai sintesis fuzzy dengan cara.

1. Jumlah nilai kolom *lower* dalam satu baris akan dibagi dengan total jumlah nilai kolom *upper*.
2. Jumlah nilai kolom *middle* dalam satu baris akan dibagi dengan total jumlah nilai kolom *middle*.
3. Jumlah nilai kolom *upper* dalam satu baris akan dibagi dengan total jumlah nilai kolom *lower*.

Berikut adalah Tabel penjumlahan pada kolom *lower*, *middle* dan *upper*.

Tabel 4.8 Penjumlahan pada kolom *lower*, *middle* dan *upper*

| Kriteria | $\sum_{j=1}^g M_i^j$ | | |
|-----------------------------------|----------------------|----------|----------|
| | <i>l</i> | <i>m</i> | <i>u</i> |
| C1 | 4,119 | 4,7333 | 5,4 |
| C2 | 2,4555 | 2,8167 | 3,4524 |
| C3 | 11,5 | 13,5 | 15 |
| C4 | 6,6667 | 8 | 10 |
| C5 | 4,25 | 4,7222 | 5,2222 |
| $\sum_{i=1}^h \sum_{j=1}^g M_i^j$ | 28,9912 | 33,7722 | 39,0746 |

Berikut ini penjelasan perhitungan perolehan nilai Tabel 4.10 untuk baris 1 menggunakan Persamaan 2.9.

$$\begin{aligned}
 \sum_{j=1}^g M_i^j &= \sum_{j=1}^g l_j, \sum_{j=1}^g m_j, \sum_{j=1}^g u_j \\
 &= \{l_1, l_2, l_3, l_4, l_5\}, \{m_1, m_2, m_3, m_4, m_5\}, \{u_1, u_2, u_3, u_4, u_5\} \\
 &= \{1+1,5+0,2857+0,3333+1\}, \{1+2+0,3333+0,4+1\}, \\
 &\quad \{1+2,5+0,4+0,5+1\} \\
 &= \{4,119, 4,7333, 5,4\}
 \end{aligned}$$

Demikian seterusnya hingga baris terakhir. Untuk perhitungan total jumlah nilai pada kolom *lower*, *middle* dan *upper* menggunakan Persamaan 2.10.

$$\begin{aligned}\sum_{i=1}^h \sum_{j=1}^g M_i^j &= \sum_{j=1}^g l_j, \sum_{j=1}^g m_j, \sum_{j=1}^g u_j \\ &= \{l_1, l_2, l_3, l_4, l_5\}, \{m_1, m_2, m_3, m_4, m_5\}, \{u_1, u_2, u_3, u_4, u_5\} \\ &= \{4,119+2,4555+11,5+6,6667+4,25\}, \\ &\quad \{4,7333+2,8167+13,5+8+4,7222\}, \\ &\quad \{5,4+3,4524+15+10+5,2222\} \\ &= \{28,9912, 33,7722, 39,0746\}\end{aligned}$$

Setelah diperoleh nilai pada Tabel 4.9 maka akan dilakukan perhitungan menentukan matriks sintesis *fuzzy* dengan contoh perhitungan untuk menentukan nilai pada baris 1 menggunakan Persamaan 2.11.

$$\begin{aligned}\text{Sintesis Fuzzy 1} &= \left\{ \frac{4,119}{28,9912} \right\}, \left\{ \frac{4,7333}{33,7722} \right\}, \left\{ \frac{5,4}{39,0746} \right\} \\ &= \{0,1054, 0,1402, 0,1863\}\end{aligned}$$

Demikian seterusnya hingga baris terakhir. Hasil dari sintesis *fuzzy* dapat dilihat pada Tabel 4.9.

Tabel 4.9 Matriks sintesis *fuzzy*

| Kriteria | Si | | |
|----------|----------|----------|----------|
| | <i>l</i> | <i>m</i> | <i>u</i> |
| SK1 | 0,1054 | 0,1402 | 0,1863 |
| SK2 | 0,0628 | 0,0834 | 0,1191 |
| SK3 | 0,2943 | 0,3997 | 0,5174 |
| SK4 | 0,1706 | 0,2369 | 0,3449 |
| Sk5 | 0,1088 | 0,1398 | 0,1801 |

4.3.8 Vektor dan Ordinat Defuzzifikasi

Menentukan Vektor dan Ordinat Defuzzifikasi dapat dilakukan setelah memperoleh matriks sintesis *fuzzy*. Dalam pemberian nilai vektor baru adalah untuk menentukan antara kriteria satu dengan semua kriteria lainnya.

Penentuan vektor C1 akan membandingkan dengan vektor yang lain dengan menggunakan Persamaan 2.12 dan Tabel 4.9. Berikut contoh perbandingan Vektor C1 dengan C2.

$$V(C_1, C_2) = \frac{0,0889-0,3333}{(0,1621-0,3333)-(0,1646-0,0889)} = 0,9899$$

dikarenakan tidak memenuhi persyaratan $m_1 \geq m_2$ ataupun $l_2 \geq u_1$.

Demikian seterusnya hingga didapat nilai perbandingan pada Tabel 4.10.

Tabel 4.10 Nilai vektor defuzzifikasi

| M2 | M1 | V(M2 > M1) |
|----|----|------------|
| C1 | C1 | 1 |
| | C2 | 1 |
| | C3 | 0 |
| | C4 | 0,1393 |
| | C5 | 1 |
| C2 | C1 | 0,1941 |
| | C2 | 1 |
| | C3 | 0 |
| | C4 | 0 |
| | C5 | 0,1546 |
| C3 | C1 | 1 |
| | C2 | 1 |
| | C3 | 1 |
| | C4 | 1 |
| | C5 | 1 |
| C4 | C1 | 1 |
| | C2 | 1 |
| | C3 | 0,2371 |
| | C4 | 1 |
| | C5 | 1 |
| C5 | C1 | 0,9956 |
| | C2 | 1 |
| | C3 | 0 |
| | C4 | 0,0893 |
| | C5 | 1 |

Tahap selanjutnya adalah penentuan ordinat defuzzifikasi dengan menggunakan Persamaan 2.13 dan Tabel 4.12. Berikut adalah penentuan ordinat defuzzifikasi pada kriteria 1(C1):

$$V(C1) = \min (1, 1, 0, 0,1393, 1).$$

$$= 0$$

Demikian dengan kriteria yang lainnya didapat nilai ordinat defuzzifikasi yang di tampilkan dalam Tabel 4.11.

Tabel 4.11 Nilai ordinat defuzzifikasi

| Kriteria | V(M) |
|----------|------|
| C1 | 0 |
| C2 | 0 |

| | |
|----|--------|
| C3 | 1 |
| C4 | 0,2371 |
| C5 | 0 |

4.3.9 Normalisasi vektor *defuzzy*

Normalisasi vektor *defuzzy* dilakukan setelah memperoleh vektor dan ordinat defuzzifikasi atau vektor *defuzzy*. Normalisasi vektor *defuzzy* atau bias disebut bobot akhir kriteria diperoleh dari pembagian antara nilai *defuzzy* dengan total jumlah nilai *defuzzy*.

Berikut adalah contoh perhitungannya menggunakan Persamaan 2.15 dan Tabel 4.11.

$$\text{Bobot kriteria 1} = \frac{0}{0+0+1+0,2371+0}$$

$$= \frac{0}{1,2371} = 0$$

Demikian seterusnya hingga kriteria terakhir diperoleh nilai bobot kriteria dapat dilihat pada Tabel 4.12.

Tabel 4.12 Bobot kriteria

| Kriteria | Bobot Kriteria |
|----------|----------------|
| C1 | 0 |
| C2 | 0 |
| C3 | 0,8083 |
| C4 | 0,1917 |
| C5 | 0 |

Bobot kriteria yang telah ditentukan akan digunakan ketahapan selanjutnya yakni dengan metode TOPSIS.

4.3.10 Matriks Keputusan

Metode TOPSIS diawali dengan pembuatan matriks keputusan. Matriks keputusan berasal dari data rumah makan di kota Malang yang akan dikonversi ke dalam nilai numerik. Berikut contoh dari data rumah makan yang pernah dikunjungi oleh pelanggan 1 dapat dilihat pada Tabel 4.1. Peneliti merubah nilai dari Data rumah makan dengan skala seperti Tabel 4.13.

Tabel 4.13 Konversi data

| No | Kriteria | Data Sebenarnya | Konversi |
|----|-------------|--------------------------|----------|
| 1 | Jumlah menu | Antara 1 sampai 5 Menu | 1 |
| | | Antara 6 sampai 10 Menu | 2 |
| | | Antara 11 sampai 15 Menu | 3 |
| | | Antara 16 sampai 20 Menu | 4 |
| | | Di atas 20 | 5 |

| | | | |
|---|--------------------|--|-----------------------|
| 2 | Rating Rumah makan | 1 2 3 4 5 | 1 2 3 4 5 |
| 3 | Harga Menu | Di atas Rp.40000 Di atas Rp.30000 sampai Rp.40000 Di atas Rp.20000 sampai Rp.30000 Di atas Rp.10000 sampai Rp.20000 Di atas Rp.0 sampai Rp.10000 | 1 2 3 4 5 |
| 4 | Jarak Rumah Makan | Di atas 4 km Di atas 3 Km sampai 4 Km Di atas 2 Km sampai 3 Km Di atas 1 Km sampai 2 Km Di atas 0 Km sampai 1 Km | 1 2 3 4 5 |
| 5 | Waktu buka | Di atas 0 Jam sampai 6 Jam Di atas 6 Jam sampai 12 Jam Di atas 12 Jam sampai 18 Jam Di atas 18 Jam | 1 2 3 4 |

Data rumah makan pada Tabel 4.1 akan dikonversi menggunakan konversi di atas dan memperoleh matriks keputusan seperti pada Tabel 4.14.

Tabel 4.14 Matriks keputusan

| Alternatif | C1 | C2 | C3 | C4 | C5 |
|------------|----|----|----|----|----|
| A2 | 5 | 4 | 4 | 5 | 3 |
| A3 | 5 | 4 | 4 | 4 | 4 |
| A5 | 5 | 4 | 3 | 2 | 1 |
| A8 | 3 | 4 | 4 | 4 | 4 |
| A10 | 2 | 4 | 4 | 2 | 3 |
| A12 | 3 | 4 | 4 | 5 | 2 |
| A13 | 5 | 4 | 4 | 5 | 2 |
| A18 | 4 | 4 | 3 | 3 | 2 |
| A20 | 4 | 4 | 4 | 1 | 2 |
| A22 | 5 | 4 | 4 | 3 | 2 |
| A23 | 3 | 4 | 3 | 1 | 2 |
| A26 | 5 | 4 | 4 | 5 | 2 |
| A28 | 4 | 4 | 4 | 3 | 2 |

| | | | | | |
|-----|---|---|---|---|---|
| A30 | 2 | 4 | 3 | 2 | 2 |
|-----|---|---|---|---|---|

4.3.11 Normalisasi Matriks Keputusan

Matriks keputusan yang dibuat akan dinormalisasi terlebih dahulu pada tahapan ini. Normalisasi matriks keputusan adalah setiap data matriks keputusan akan dibagi dengan jumlah data matriks keputusan pangkat 2 pada kolom yang sama. Berikut contoh perhitungan normalisasi matriks pada baris 1 kolom 1 dengan acuan data pada Tabel 4.14 dengan menggunakan Persamaan 2.16.

$$\text{Normalisasi} = \frac{5}{\sqrt{(5^2)+(5^2)+(5^2)+(3^2)+(2^2)+(3^2)+(5^2)+(4^2)+(4^2)+(5^2)+(3^2)+(5^2)(4^2)+(2^2)}}$$

$$= 0,3276$$

Demikian seterusnya hingga baris dan kolom terakhir pada matriks. Hasil dari normalisasi matriks keputusan dapat dilihat pada Tabel 4.15.

Tabel 4.15 Matriks normalisasi keputusan

| Alternatif | C1 | C2 | C3 | C4 | C5 |
|------------|--------|--------|--------|--------|--------|
| A2 | 0,3276 | 0,2673 | 0,2857 | 0,3801 | 0,3216 |
| A3 | 0,3276 | 0,2673 | 0,2857 | 0,3041 | 0,4288 |
| A5 | 0,3276 | 0,2673 | 0,2143 | 0,1521 | 0,1072 |
| A8 | 0,1965 | 0,2673 | 0,2857 | 0,3041 | 0,4288 |
| A10 | 0,1310 | 0,2673 | 0,2857 | 0,1521 | 0,3216 |
| A12 | 0,1965 | 0,2673 | 0,2857 | 0,3801 | 0,2144 |
| A13 | 0,3276 | 0,2673 | 0,2857 | 0,3801 | 0,2144 |
| A18 | 0,2620 | 0,2673 | 0,2143 | 0,2281 | 0,2144 |
| A20 | 0,2620 | 0,2673 | 0,2857 | 0,0760 | 0,2144 |
| A22 | 0,3276 | 0,2673 | 0,2857 | 0,2281 | 0,2144 |
| A23 | 0,1965 | 0,2673 | 0,2143 | 0,0760 | 0,2144 |
| A26 | 0,3276 | 0,2673 | 0,2857 | 0,3801 | 0,2144 |
| A28 | 0,2620 | 0,2673 | 0,2857 | 0,2281 | 0,2144 |
| A30 | 0,1310 | 0,2673 | 0,2143 | 0,1521 | 0,2144 |

4.3.12 Matriks Normalisasi Terbobot

Matriks keputusan yang dinormalisasi akan digunakan untuk pembuatan matriks normalisasi terbobot. Matriks normalisasi terbobot dapat diperoleh dari perkalian antara data matriks normalisasi dengan data bobot kriteria terakhir yang telah diperoleh dari proses FAHP sebelumnya sesuai kriteria masing-masing. Berikut contoh normalisasi terbobot data matriks pada baris 1 kolom 1 dengan acuan matriks normalisasi keputusan Tabel 4.17 terhadap bobot kriteria pada Tabel 4.14 dengan menggunakan Persamaan 2.16.

$$\text{Normalisasi terbobot} = 0 * 0,3276 = 0$$

Demikian seterusnya hingga baris dan kolom terakhir dari matriks. Hasil matriks normalisasi terbobot dapat dilihat pada Tabel 4.16 dengan menggunakan Persamaan 2.17.

Tabel 4.16 Matriks normalisasi terbobot

| Alternatif | C1 | C2 | C3 | C4 | C5 |
|------------|----|----|--------|--------|----|
| A2 | 0 | 0 | 0,2309 | 0,0729 | 0 |
| A3 | 0 | 0 | 0,2309 | 0,0583 | 0 |
| A5 | 0 | 0 | 0,1732 | 0,0291 | 0 |
| A8 | 0 | 0 | 0,2309 | 0,0583 | 0 |
| A10 | 0 | 0 | 0,2309 | 0,0291 | 0 |
| A12 | 0 | 0 | 0,2309 | 0,0729 | 0 |
| A13 | 0 | 0 | 0,2309 | 0,0729 | 0 |
| A18 | 0 | 0 | 0,1732 | 0,0437 | 0 |
| A20 | 0 | 0 | 0,2309 | 0,0146 | 0 |
| A22 | 0 | 0 | 0,2309 | 0,0437 | 0 |
| A23 | 0 | 0 | 0,1732 | 0,0146 | 0 |
| A26 | 0 | 0 | 0,2309 | 0,0729 | 0 |
| A28 | 0 | 0 | 0,2309 | 0,0437 | 0 |
| A30 | 0 | 0 | 0,1732 | 0,0291 | 0 |

4.3.13 Solusi Ideal Positif dan Solusi Ideal Negatif

Matriks normalisasi terbobot digunakan untuk mencari nilai solusi ideal positif dan solusi ideal negatif. Solusi ideal positif diperoleh dari nilai maksimal pada matriks normalisasi terbobot pada kolom yang sama. Solusi ideal negatif diperoleh dari nilai minimal pada matriks normalisasi terbobot pada kolom yang sama. Berikut ini contoh perhitungan nilai solusi ideal positif dan solusi ideal negatif dengan acuan Tabel 4.18 pada kolom 1 dengan menggunakan Persamaan 2.18 dan 2.19.

$$A^+ = \max(0, \dots, 0)$$

$$= 0$$

$$A^- = \min(0, \dots, 0)$$

$$= 0$$

Demikian seterusnya hingga kriteria terakhir. Hasil akhirnya didapat nilai matriks ideal positif-negatif sesuai Tabel 4.17.

Tabel 4.17 Matriks ideal positif-negatif

| | C1 | C2 | C3 | C4 | C5 |
|----|----|----|--------|--------|----|
| A+ | 0 | 0 | 0,2309 | 0,0729 | 0 |
| A- | 0 | 0 | 0,1732 | 0,0146 | 0 |

4.3.14 Jarak Solusi Ideal Positif dan Jarak Solusi Ideal Negatif

Matriks ideal positif-negatif digunakan untuk mencari nilai jarak solusi ideal positif dan jarak solusi ideal negatif pada tahapan ini. Jarak solusi ideal positif didapat dari akar jumlah pengurangan data matriks normalisasi terbobot pada satu baris dengan matriks ideal positif-negatif sesuai kolom yang sama pada baris A+ dipangkat 2. Jarak solusi ideal negatif didapat dari akar jumlah pengurangan data matriks normalisasi terbobot pada satu baris dengan matriks ideal positif-negatif sesuai kolom yang sama pada baris A- dipangkat 2. Berikut ini contoh perhitungan nilai jarak solusi ideal positif dan jarak solusi ideal negatif dengan acuan Tabel 4.18 pada kolom 1 dan Tabel 4.19 dengan menggunakan Persamaan 2.20 dan Persamaan 2.21.

$$\begin{aligned}
 D+ &= \sqrt{(0-0)^2 + (0-0)^2 + (0,2309-0,2309)^2 + (0,0729-0,0146)^2 + (0-0)^2} \\
 &= \sqrt{0+0+0+0+0} \\
 &= \sqrt{0} = 0 \\
 D- &= \sqrt{(0-0)^2 + (0-0)^2 + (0,2309-0,1732)^2 + (0,0729-0,0902)^2 + (0-0)^2} \\
 &= \sqrt{0+0+0,0033+0,0034+0} \\
 &= \sqrt{0,0067} = 0,082
 \end{aligned}$$

Demikian hingga baris dan kolom terakhir matriks normalisasi terbobot. Hasil akhir dapat dilihat pada Tabel 4.18.

Tabel 4.18 Matriks jarak ideal positif-negatif

| Alternatif | D+ | D- |
|------------|--------|--------|
| A2 | 0 | 0,0820 |
| A3 | 0,0146 | 0,0724 |
| A5 | 0,0724 | 0,0146 |
| A8 | 0,0146 | 0,0724 |
| A10 | 0,0437 | 0,0595 |
| A12 | 0 | 0,0820 |
| A13 | 0 | 0,0820 |
| A18 | 0,0647 | 0,0291 |
| A20 | 0,0583 | 0,0577 |
| A22 | 0,0291 | 0,0647 |
| A23 | 0,0820 | 0 |
| A26 | 0 | 0,0820 |
| A28 | 0,0291 | 0,0647 |
| A30 | 0,0724 | 0,0146 |

4.3.15 Nilai *Preference*

Matriks jarak ideal positif-negatif digunakan untuk mencari nilai *preference* masing-masing alternatif pada tahapan ini. Nilai *preference* didapat dari pembagian antara jarak solusi ideal positif dengan penjumlahan jarak ideal solusi positif dan jarak ideal solusi negatif. Berikut ini contoh perhitungan nilai *preference* dengan acuan Tabel 4.18 dengan Persamaan 2.22 pada kolom 1.

$$\begin{aligned} \text{preference} &= \frac{0,082}{(0+0,082)} \\ &= \frac{0,082}{0,082} \\ &= 1 \end{aligned}$$

Demikian seterusnya hingga baris terakhir dari matriks jarak ideal positif-negatif. Hasil akhir dari proses ini didapat daftar nilai *preference* masing-masing alternative sesuai Tabel 4.19.

Tabel 4.19 Nilai *preference*

| Alternatif | <i>Preference</i> |
|------------|-------------------|
| A2 | 1 |
| A3 | 0,8325 |
| A5 | 0,1675 |
| A8 | 0,8325 |
| A10 | 0,5766 |
| A12 | 1 |
| A13 | 1 |
| A18 | 0,3107 |
| A20 | 0,4976 |
| A22 | 0,6893 |
| A23 | 0 |
| A26 | 1 |
| A28 | 0,6893 |
| A30 | 0,1675 |

Setelah diketahui nilai *preference* untuk pelanggan 1 maka dilakukan untuk Pengurutan nilai *preference* yang akan dijadikan rekomendasi rumah makan kepada pelanggan 1. Pengurutan nilai *preference* dapat dilihat pada Tabel 4.20.

Tabel 4.20 Pengurutan nilai *preference*

| Alternatif | D |
|------------|--------|
| A2 | 1 |
| A26 | 1 |
| A12 | 1 |
| A13 | 1 |
| A8 | 0,8325 |

| | |
|-----|--------|
| A3 | 0,8325 |
| A28 | 0,6893 |
| A22 | 0,6893 |
| A10 | 0,5766 |
| A20 | 0,4976 |
| A18 | 0,3107 |
| A30 | 0,1675 |
| A5 | 0,1675 |
| A23 | 0 |

Hasil rekomendasi untuk pelanggan 1 menggunakan metode FAHP-TOPSIS adalah A2 (Ayam Bawang cak Per(suhat)), A26 (Ayam D'keprek Kertosentono) , A12 (Mie Jogging), A13 (Kober Mie Setan Soekarno Hatta), A8 (Ayam Uleg Cak Abit(Sumbersari)), A3 (Ayam Goreng Nelongso (Suhat)), A28 (Burger Buto 27), A22 (Rumah Makan Dapoer Mie Galau), A10 (Soto Ambengan), A20 (Depot Pangsit Mie Bromo Pojok), A18 (Jank Jank Wings, Klojen), A30 (De Chicken, Blimbing), A5 (Ikana), A23 (Waroeng Ayam Bakar Lientang(Sawojajar)).

4.4 Perancangan Pengujian

Perancangan ini dilakukan untuk melakukan pengujian terhadap algoritme FAHP-TOPSIS. Pengujian yang akan dilakukan untuk mengetahui tingkat akurasi antara hasil pemeringkatan dari sistem dengan pemeringkatan pakar secara langsung dari Pelanggan. Pengujian dilakukan berupa pengujian Korelasi Spearman dimana r_X adalah peringkat yang dihasilkan sistem dan r_Y adalah peringkat yang di pilih pelanggan secara pakar secara langsung. Kolom d adalah selisih dari nilai r_X dengan r_Y dan kolom D adalah kuadrat dari d . Setelah itu nilai D total secara keseluruhan dan di tentukan nilai p dengan menggunakan Persamaan 2.24. Berikut hasil dari pemeringkatan menggunakan metode FAHP-TOPSIS terhadap pemeringkatan pakar secara langsung pada pelanggan 1 beserta nilai d dan D dapat dilihat pada Tabel 4.21.

Tabel 4.21 Hasil uji pelanggan pertama

| No | Nama Rumah makan | r_X | r_Y | d | D |
|----|--------------------------------|-------|-------|-------|--------|
| 1 | Ayam D'keprek Kertosentono | 2,5 | 2 | 0,5 | 0,25 |
| 2 | Mie Jogging | 2,5 | 4 | -1,5 | 2,25 |
| 3 | Ayam Bawang cak Per(suhat) | 2,5 | 5 | -2,5 | 6,25 |
| 4 | Kober Mie Setan Soekarno Hatta | 2,5 | 13 | -10,5 | 110,25 |
| 5 | Ayam Goreng Nelongso (Suhat) | 5,5 | 1 | 4,5 | 20,25 |
| 6 | Ayam Uleg Cak Abit(Sumbersari) | 5,5 | 6 | -0,5 | 0,25 |
| 7 | Rumah Makan Dapoer Mie Galau | 7,5 | 10 | -2,5 | 6,25 |
| 8 | Burger Buto 27 | 7,5 | 11 | -3,5 | 12,25 |
| 9 | Soto Ambengan | 9 | 7 | 2 | 4 |
| 10 | Depot Pangsit Mie Bromo Pojok | 10 | 14 | -4 | 16 |

| | | | | | |
|----|--|------|----|-----|-------|
| 11 | Jank Jank Wings, Klojen | 11 | 8 | 3 | 9 |
| 12 | De Chicken, Blimbing | 12,5 | 3 | 9,5 | 90,25 |
| 13 | Ikana | 12,5 | 12 | 0,5 | 0,25 |
| 14 | Waroeng Ayam Bakar Lientang(Sawojajar) | 14 | 9 | 5 | 25 |

Hasil D dari Tabel 4.2 akan digunakan dalam penentuan nilai korelasi Spearman pada pelanggan 1 dengan menggunakan Persamaan 2.23.

$$p = 1 - \frac{6((2,5-2)^2 + (2,5-4)^2 + \dots + (14-9)^2)}{17^3 - 17} = 0,3352$$

Hasil p yang didapat akan diinterpretasikan ke dalam tabel D.A. de Vaus pada Tabel 2,5 menghasilkan hubungan moderat.

4.5 Perancangan Antarmuka

Perancangan antarmuka yang akan dibangun meliputi menu input kuesioner, menu input data rumah makan, proses FAHP-TOPSIS serta Hasil rekomendasi rumah makan.

4.5.1 Perancangan Antarmuka Menu Input Data Pelanggan

Perancangan antarmuka menu input data pelanggan. Tampilan berisi *form* data nama, lokasi, pilihan rumah makan pelanggan dan matriks kriteria perbandingan pelanggan. Perancangan antarmuka menu input data pelanggan dapat dilihat pada Gambar 4.18.

Gambar 4.18 Perancangan antarmuka menu input data pelanggan

Keterangan:

1. Judul Aplikasi.
2. Navigasi Menu terdiri dari menu input data pelanggan, input data rumah makan, FAHP dan TOPSIS
3. *Dropdown* nama pelanggan
4. *Form* input nama pelanggan baru.
5. *Form* nama lokasi pelanggan.
6. Tombol Pencarian lokasi pelanggan.
7. Tabel Matriks Kriteria Perbandingan.
8. Tabel Matriks rumah makan yang pernah dikunjungi.
9. Tombol simpan data pelanggan.

4.5.2 Perancangan Antarmuka Menu Input Data Rumah Makan

Perancangan antarmuka menu input data rumah makan. Tampilan berisi *form* nama, lokasi, banyak menu, rata-rata harga, rating dan waktu buka rumah makan. Perancangan antarmuka menu input data rumah makan dapat dilihat pada Gambar 4.19.

Gambar 4.19 Perancangan antarmuka menu input data rumah makan

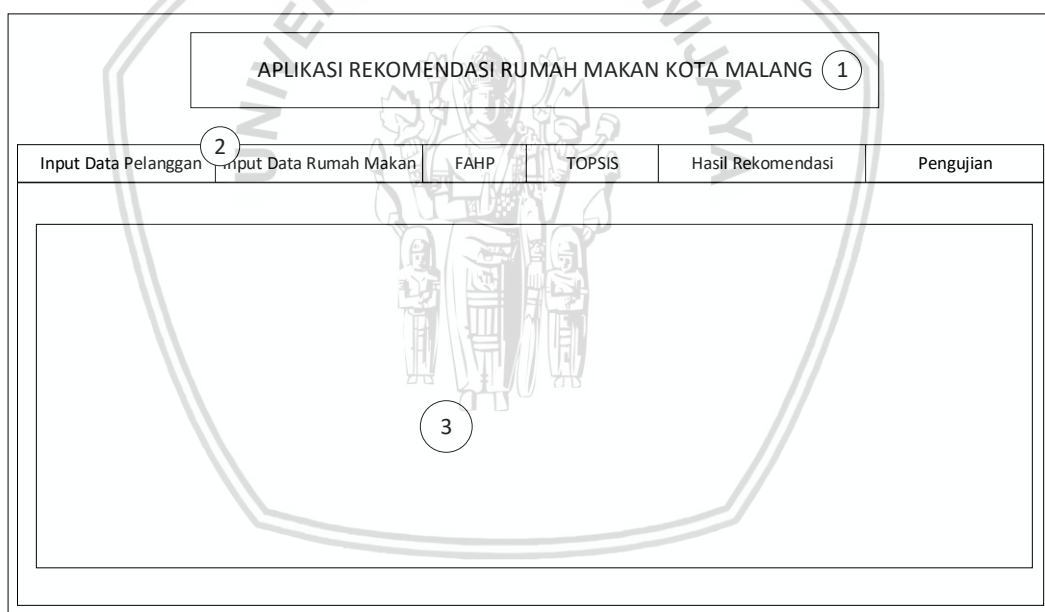
Keterangan:

1. Judul Aplikasi.
2. Navigasi Menu terdiri dari menu input data pelanggan, input data rumah makan, FAHP dan TOPSIS.
3. *Dropdown* nama rumah makan.

4. *Form* input nama rumah makan baru.
5. *Form* nama lokasi pelanggan.
6. Tombol Pencarian lokasi pelanggan.
7. *Form* input banyak menu rumah makan.
8. *Form* input rata-rata harga menu rumah makan.
9. *Form* input rating rumah makan.
10. *Form* input waktu buka rumah makan.
11. Tombol simpan data rumah makan.

4.5.3 Perancangan Antarmuka Menu FAHP, TOPSIS, Rekomendasi dan Pengujian

Perancangan antarmuka menu FAHP, TOPSIS, Rekomendasi dan Pengujian. Tampilan berisi dan TOPSIS dapat dilihat pada Gambar 4.20.



Gambar 4.20 Perancangan antarmuka menu fahp, topsis, hasil rekomendasi dan pengujian

Keterangan:

1. Judul Aplikasi.
2. Navigasi Menu terdiri dari menu input data pelanggan, input data rumah makan, FAHP dan TOPSIS
3. Tombol menampilkan proses FAHP, TOPSIS, hasil rekomendasi dan pengujian.

4.6 Implementasi algoritme FAHP-TOPSIS

Pada tahap ini menjelaskan implementasi algoritme FAHP-TOPSIS ke dalam bahasa pemrograman menggunakan framework Code Igniter. Terdapat implementasi kode-kode pemrograman mengenai Inisialisasi matriks perbandingan kriteria, normalisasi matriks perbandingan kriteria, bobot prioritas, Eigen maks, *consistency ratio*, matriks perbandingan kriteria dengan skala tfn, matriks sintesis *fuzzy*, vektor dan ordinat defuzzifikasi, normalisasi vektor *fuzzy*, normalisasi matriks keputusan, matriks normalisasi terbobot, solusi ideal positif dan solusi ideal negatif, jarak solusi ideal positif dan jarak solusi ideal negatif, dan nilai *preference*.

4.6.1 Implementasi Matriks Perbandingan Kriteria

Proses pembuatan matriks kriteria perbandingan adalah proses pembuatan tabel matriks kriteria perbandingan dari *database*. Implementasi matriks kriteria perbandingan dapat dilihat pada Kode Program 4.1.

```

1  function make_matrix($count,$value_criteria){
2      $nVCriteria = array();
3      $SumVCriteria = array();
4      foreach($value_criteria as $rows ){
5          $row = $rows->row;
6          $column = $rows->col;
7          $nVCriteria[$row][$column]=$rows->value;
8      }
9      for ($i = 0; $i < $count; $i++) {
10         $SumVCriteria[$i] = 0;
11         for ($j = 0; $j < $count; $j++) {
12             $SumVCriteria[$i] += $nVCriteria[$j][$i];
13         }
14     }
15     return array('nV' => $nVCriteria,
16                 'sumV' => $SumVCriteria);
17 }

```

Kode Program 4.1 Matriks kriteria perbandingan

Penjelasan kode program:

1. Baris 1: Nama fungsi *make_matrix* dengan input parameter *\$count* dengan nilai banyak kriteria dan *\$value_criteria* dengan nilai kriteria perbandingan dari *database*
2. Baris 2-3 mendefinisikan inisialisasi awal untuk *array nVCriteria* untuk nilai kriteria dan *array \$SumVCriteria* jumlah nilai masing-masing kriteria dalam 1 kolom tabel.
3. Baris 4-8: merupakan perulangan untuk memberi nilai pada *array \$nVCriteria* sesuai dengan perbandingan kriteria dari *database*.
4. Baris 9-14: merupakan perulangan untuk memberi nilai pada *array \$SumVCriteria* berdasarkan jumlah nilai masing-masing kriteria dalam 1 kolom tabel.
5. Baris 15 : merupakan *return value* dari fungsi *make_matrix* berupa array *index "nV"* bernilai *\$nVCriteria* dan *index "sumV"* bernilai *\$ SumVCriteria*

4.6.2 Implementasi Normalisasi Matriks Perbandingan Kriteria

Proses normalisasi matriks kriteria perbandingan adalah proses normalisasi dari matriks kriteria perbandingan dari *database*. Implementasi normalisasi matriks kriteria perbandingan dapat dilihat pada Kode Program 4.2.

```

1  function normal_matrix($count,$nVCriteria,$SumVCriteria){
2      $NormalVCriteria = array();
3      $SumRowVCriteria = array();
4      $WVCriteria = array();
5      for ($i = 0; $i < $count; $i++) {
6          $SumRowVCriteria[$i] = 0;
7          for ($j = 0; $j < $count; $j++) {
8              $NormalVCriteria[$i][$j] =
9                  $nVCriteria[$i][$j]/$SumVCriteria[$j];
10             }
11             for ($k = 0; $k < $count; $k++) {
12                 $SumRowVCriteria[$i] +=
13                     $NormalVCriteria[$i][$k];
14             }
15             $WVCriteria[$i] = $SumRowVCriteria[$i]/$count;
16         }
17         return array('NormalV' => $NormalVCriteria,
18                     'SumV' => $SumRowVCriteria,
19                     'WV' => $WVCriteria);
20     }
21 }

```

Kode Program 4.2 Normalisasi matriks kriteria perbandingan

Penjelasan kode program:

1. Baris 1-3: mendefinisikan inisialisasi awal untuk *array* *\$NormalVCriteria* untuk nilai normalisasi kriteria perbandingan, *array* *\$SumRowVCriteria* jumlah nilai masing-masing kriteria dalam 1 baris tabel dan *array* *\$WVCriteria* untuk nilai bobot awal masing-masing kriteria.
2. Baris 4-6: merupakan perulangan untuk memberi nilai pada *array* *\$SumRowVCriteria* dengan nilai awal 0.
3. Baris 7-15: merupakan perulangan untuk memberi nilai *array* *\$NormalVCriteria* dari *\$nVCriteria* dibagi *\$SumVCriteria*, *\$SumRowVCriteria* dengan jumlah nilai *\$NormalVCriteria* yang telah didapat sesuai kriteria masing-masing dalam 1 baris tabel dan *\$WVCriteria* dari nilai *\$SumRowVCriteria* dibagi dengan *\$count*.

4.6.3 Implementasi Vektor Bobot

Proses ini bertujuan untuk memperoleh nilai dari vektor bobot. Implementasi inisialisasi vektor bobot dapat dilihat pada Kode Program 4.3.

```

1  function vector_criteria($count,$nVCriteria,$WVCriteria){
2      $VectorCriteria = array();
3      for ($i = 0; $i < $count; $i++) {
4          $VectorCriteria[$i] = 0;
5          for($j=0;$j< $count; $j++){
6              $VectorCriteria[$i] +=
7                  ($nVCriteria[$i][$j]*$WVCriteria[$j]);
8          }
9      }
10     return $VectorCriteria;

```

| | |
|----|---|
| 11 | } |
|----|---|

Kode Program 4.3 Vektor bobot

Penjelasan kode program:

1. Baris 1: mendefinisikan inisialisasi awal untuk *array \$VectorCriteria* yang akan digunakan sebagai bobot prioritas.
2. Baris 2-7: merupakan perulangan untuk memberi nilai pada *array \$VectorCriteria* dengan menjumlahkan dari perkalian dari nilai kriteria pada variabel *\$nVCriteria* pada proses inisialisasi matriks kriteria perbandingan terhadap bobot prioritas awal pada variabel *\$WVCriteria* pada proses normalisasi matriks kriteria perbandingan.
3. Baris 7-15: merupakan perulangan untuk memberi nilai pada *array \$NormalVCriteria* dari nilai *\$nVCriteria* dibagi *\$SumVCriteria* pada proses sebelumnya, *\$SumRowVCriteria* dengan jumlah nilai *\$NormalVCriteria* yang telah didapat sesuai kriteria masing-masing dalam 1 baris tabel dan *\$WVCriteria* dari nilai *\$SumRowVCriteria* dibagi dengan *\$count* (banyaknya kriteria yang digunakan).

4.6.4 Implementasi Nilai Prioritas

Proses ini bertujuan untuk memperoleh nilai prioritas. Implementasi nilai prioritas pada Kode Program 4.4.

| | |
|---|---|
| 1 | Function priority_criteria(\$count, \$VectorCriteria,\$WVCriteria){ |
| 2 | \$PriorityVCriteria = array(); |
| 3 | for (\$i = 0; \$i < \$count; \$i++){ |
| 4 | \$PriorityVCriteria[\$i] = |
| 5 | \$VectorCriteria[\$i]/\$WVCriteria[\$i]; |
| 6 | } |
| 7 | return \$PriorityVCriteria; |
| 8 | } |

Kode Program 4.4 Nilai prioritas

Penjelasan kode program:

1. Baris 1-2: mendefinisikan inisialisasi awal untuk *array \$PriorityVCriteria* yang sebagai tempat penyimpanan nilai prioritas dan *\$lamdaMax* dengan nilai awal 0.
2. Baris 3-5: merupakan perulangan untuk memberi nilai pada *array \$PriorityVCriteria* dengan membagi nilai bobot prioritas pada variabel *\$VectorCriteria* terhadap Bobot Kriteria Awal pada variabel *\$WVCriteria*.
3. Baris 6-9: merupakan penentuan nilai *\$lamdaMax* dengan perulangan pada baris 6-8 untuk menjumlahkan nilai prioritas pada Variabel *\$PriorityVCriteria*. Hasil dari penjumlahan tersebut akan dibagi dengan banyaknya kriteria pada variabel *\$count* untuk memperoleh nilai *\$lamdaMax*.
4. Baris 10-11: merupakan penentuan nilai *Consistency Ratio* atau *\$CR* dengan menenukan nilai *Consistency Index* atau *\$CI* terlebih dahulu. *\$CI*

didapat dari baris program ke 10 dan $\$CR$ didapat dari baris program ke 11.

4.6.5 Implementasi Pengecekan Konsistensi

Proses ini bertujuan untuk memperoleh pengecekan apakah matriks perbandingan kriteria yang diperoleh sudah konsisten atau belum. Implementasi Pengecekan Konsistensi pada Kode Program 4.5.

```

1  function check_CR($count,$PriorityVCriteria){
2      $lamdaMax = 0;
3      for ($i = 0; $i < $count; $i++) {
4          $lamdaMax += $PriorityVCriteria[$i];
5      }
6      $lamdaMax = $lamdaMax/$count;
7      $CI = ($lamdaMax-$count)/($count-1);
8      $CR = $CI/1.12;
9
10     return array('lamdaMax' => $lamdaMax,
11                  'CI' => $CI,
12                  'CR' => $CR);
13 }

```

Kode Program 4.5 Pengecekan konsistensi

Penjelasan kode program:

1. Baris 1-2: mendefinisikan inialisasi awal untuk *array* $\$PriorityVCriteria$ yang sebagai tempat penyimpanan nilai prioritas dan $\$lamdaMax$ dengan nilai awal 0.
2. Baris 3-5: merupakan perulangan untuk memberi nilai pada *array* $\$PriorityVCriteria$ dengan membagi nilai bobot prioritas pada variabel $\$VectorCriteria$ terhadap Bobot Kriteria Awal pada variabel $\$WVCriteria$.
3. Baris 6-9: merupakan penentuan nilai $\$lamdaMax$ dengan perulangan pada baris 6-8 untuk menjumlahkan nilai prioritas pada Variabel $\$PriorityVCriteria$. Hasil dari penjumlahan tersebut akan dibagi dengan banyaknya kriteria pada variabel $\$count$ untuk memperoleh nilai $\$lamdaMax$.
4. Baris 10-11: merupakan penentuan nilai *Consistency Ratio* atau $\$CR$ dengan menenukan nilai *Consistency Index* atau $\$CI$ terlebih dahulu. $\$CI$ didapat dari baris program ke 10 dan $\$CR$ didapat dari baris program ke 11.

4.6.6 Implementasi Konversi Matriks Kriteria skala TFN

Matriks kriteria perbandingan disebut konsisten ketika nilai CR kurang dari 0,1. Ketika matriks kriteria perbandingan disebut konsisten maka tahap selanjutnya adalah konversi matriks kriteria perbandingan dengan tabel. Implementasi konversi matriks kriteria skala TFN pada Kode Program 4.6.

```

1  function conversion_TFN($count,$value_criteria){
2      $lowerTFN = array();
3      $middleTFN = array();
4      $upperTFN = array();
5      for($i=0;$i<$count;$i++){

```



```

6         for($j=0;$j<$count;$j++){
7             $lowerTFN[$i][$j]=1;
8             $middleTFN[$i][$j]=1;
9             $upperTFN[$i][$j]=1;
10        }
11    }
12    foreach($value_criteria as $rows ){
13        $row = $rows->row;
14        $column = $rows->col;
15        if($rows->value >= 1 ){
16            if($rows->value==1){
17                $lowerTFN[$row][$column]*=1;
18                $middleTFN[$row][$column]*=1;
19                $upperTFN[$row][$column]*=1;
20                $lowerTFN[$column][$row]*=1;
21                $middleTFN[$column][$row]*=1;
22                $upperTFN[$column][$row]*=1;
23            }elseif($rows->value==9){
24                $lowerTFN[$row][$column]*=4;
25                $middleTFN[$row][$column]*=9/2;
26                $upperTFN[$row][$column]*=9/2;
27                $lowerTFN[$column][$row]*=1/4;
28                $middleTFN[$column][$row]*=2/9;
29                $upperTFN[$column][$row]*=2/9;
30            }else{
31                $lowerTFN[$row][$column]*=
32                    ($rows->value-1)/2;
33                $middleTFN[$row][$column]*=
34                    $rows->value/2;
35                $upperTFN[$row][$column]*=
36                    ($rows->value+1)/2;
37                $lowerTFN[$column][$row]*=
38                    2/($rows->value+1);
39                $middleTFN[$column][$row]*=
40                    2/($rows->value);
41                $upperTFN[$column][$row]*=
42                    2/($rows->value-1);
43            }
44        }
45    }
46    return array ('lower'=>$lowerTFN,
47                'middle'=>$middleTFN,
48                'upper'=>$upperTFN);
49 }

```

Kode Program 4.6 Konversi matriks kriteria skala TFN

Penjelasan kode program:

1. Baris 1-3: mendefinisikan inisialisasi awal untuk *array* *\$lowerTFN* yang sebagai tempat penyimpanan nilai bawah atau *lower*, *array* *\$middleTFN* yang sebagai tempat penyimpanan nilai tengah atau *middle* dan *array* *\$upperTFN* yang sebagai tempat penyimpanan nilai atas atau *upper* dari konversi nilai matriks perbandingan.
2. Baris 4-10: merupakan perulangan untuk memberi inisialisasi nilai awal pada variabel *\$lowerTFN*, *\$middleTFN* dan *\$upperTFN*.
3. Baris 11-39: merupakan penentuan nilai pada masing-masing variabel *\$lowerTFN*, *\$middleTFN* serta *\$upperTFN* dari hasil konversi nilai matriks kriteria perbandingan diawal.

4.6.7 Implementasi Sintesis Fuzzy

Proses ini dilakukan setelah pengkonfersian matriks kriteria perbandingan dengan tabel TFN. Implementasi sintesis *fuzzy* pada Kode Program 4.7.

```

1  function synthesis_fuzzy($count,$arrayTFN){
2      $SumRowLowerCriteria = array();
3      $SumRowMiddleCriteria = array();
4      $SumRowUpperCriteria = array();
5      for ($i = 0; $i < $count; $i++) {
6          $SumRowLowerCriteria[$i]=0;
7          $SumRowMiddleCriteria[$i]=0;
8          $SumRowUpperCriteria[$i]=0;
9          for ($j = 0; $j < $count; $j++) {
10             $SumRowLowerCriteria[$i]+
11             $arrayTFN['lower'][$i][$j];
12             $SumRowMiddleCriteria[$i]+
13             $arrayTFN['middle'][$i][$j];
14             $SumRowUpperCriteria[$i]+
15             $arrayTFN['upper'][$i][$j];
16         }
17     }
18     $AllSumLowerCriteria =
19     array_sum($SumRowLowerCriteria);
20     $AllSumMiddleCriteria =
21     array_sum($SumRowMiddleCriteria);
22     $AllSumUpperCriteria =
23     array_sum($SumRowUpperCriteria);
24     $SinFLower = array();
25     $SinFMiddle = array();
26     $SinFUpper = array();
27     for ($i = 0; $i < $count; $i++) {
28         for ($j = 0; $j < $count; $j++) {
29             $SinFLower[$i]=
30             $SumRowLowerCriteria[$i]/
31             $AllSumUpperCriteria;
32             $SinFMiddle[$i]=
33             $SumRowMiddleCriteria[$i]/
34             $AllSumMiddleCriteria;
35             $SinFUpper[$i]=
36             $SumRowUpperCriteria[$i]/
37             $AllSumLowerCriteria;
38         }
39     }
40     return array ('SumRLower' => $SumRowLowerCriteria,
41                 'SumRMiddle' => $SumRowMiddleCriteria,
42                 'SumRUpper' => $SumRowUpperCriteria,
43                 'SinFLower' => $SinFLower,
44                 'SinFMiddle' => $SinFMiddle,
45                 'SinFUpper' => $SinFUpper);
46 }

```

Kode Program 4.7 Sintesis fuzzy

Penjelasan kode program:

1. Baris 1-3: mendefinisikan inisialisasi awal untuk *array* *\$SumRowLowerCriteria* sebagai tempat penyimpanan penjumlahan nilai *lower* pada masing-masing kriteria dalam 1 baris tabel, *array* *\$SumRowMiddleCriteria* sebagai tempat penyimpanan penjumlahan nilai *middle* pada masing-masing kriteria dalam 1 baris tabel dan *array* *\$SumRowUpperCriteria* sebagai tempat penyimpanan penjumlahan nilai *upper* pada masing-masing kriteria dalam 1 baris tabel.

2. Baris 4-13: merupakan perulangan untuk menentukan penjumlahan nilai *lower* pada masing-masing kriteria dalam 1 baris tabel pada variabel *\$SumRowLowerCriteria*, penjumlahan nilai *middle* pada masing-masing kriteria dalam 1 baris tabel pada variabel *\$SumRowMiddleCriteria*, penjumlahan nilai *upper* pada masing-masing kriteria dalam 1 baris tabel pada variabel *\$SumRowUpperCriteria*.
3. Baris 14-16: merupakan penjumlahan semua nilai *lower* dari array *\$SumRowLowerCriteria* pada variabel *\$AllSumLowerCriteria*, penjumlahan semua nilai *middle* dari array *\$SumRowMiddleCriteria* pada variabel *\$AllSumMiddleCriteria* dan penjumlahan semua nilai *upper* dari array *\$SumRowUpperCriteria* pada variabel *\$AllSumUpperCriteria*.
4. Baris 17-19: mendefinisikan inisialisasi awal untuk array *\$SinFLower* sebagai tempat penyimpanan sintesis fuzzy *lower* masing-masing kriteria, array *\$SinFMiddle* sebagai tempat penyimpanan sintesis fuzzy *middle* masing-masing kriteria dan array *\$SinFUpper* sebagai tempat penyimpanan sintesis fuzzy *upper* masing-masing kriteria.
5. Baris 20-26: merupakan penentuan nilai sintesis fuzzy *lower* masing-masing kriteria pada array *\$SinFLower*, nilai sintesis fuzzy *middle* masing-masing kriteria pada array *\$SinFMiddle* dan nilai sintesis fuzzy *upper* masing-masing kriteria pada array *\$SinFUpper*.

4.6.8 Implementasi Vektor dan Ordinat Defuzzifikasi

Proses vektor dan ordinat defuzzifikasi dapat dilakukan setelah memperoleh matriks sintesis fuzzy. Implementasi vektor dan ordinat defuzzifikasi pada Kode Program 4.8.

```

1 function vector_ordo_defuzzy($count,$synthesisFuzzy){
2     $EqVector = array();
3     $VecWeight = array();
4     for($i=0;$i<$count;$i++){
5         for($j=0;$j<=$count;$j++){
6             if($j < $count){
7                 if($synthesisFuzzy['SinFMiddle'][$i]
8                     >= $synthesisFuzzy['SinFMiddle'][$j]){
9                     $EqVector[$i][$j] = 1;
10                }elseif($synthesisFuzzy['SinFLower'][$j]
11                    >=$synthesisFuzzy['SinFUpper'][$i]){
12                    $EqVector[$i][$j] = 0;
13                }else{
14                    $EqVector[$i][$j] =
15                        ($synthesisFuzzy['SinFLower'][$j]
16                        -$synthesisFuzzy['SinFUpper'][$i])/
17                        (($synthesisFuzzy['SinFMiddle'][$i]
18                        -$synthesisFuzzy['SinFUpper'][$i]) -
19                        ($synthesisFuzzy['SinFMiddle'][$j]
20                        -$synthesisFuzzy['SinFLower'][$j]));
21                }
22            }else{
23                $VecWeight[$i] = $EqVector[$i][0];
24                for($k=1;$k<$count;$k++){
25                    if($VecWeight[$i]>=$EqVector[$i][$k]){
26                        $VecWeight[$i] = $EqVector[$i][$k];
27                    }
28                }
29            }
30        }
31    }
32 }

```

```

28         }
29     }
30 }
31 }
32 return array ('EqVector' => $EqVector,
33             'VecWeight' => $VecWeight);
34 }

```

Kode Program 4.8 Vektor dan ordinat defuzzifikasi

Penjelasan kode program:

1. Baris 1-3: mendefinisikan inisialisasi awal untuk *array* *\$EqVector* sebagai tempat penyimpanan nilai vektor perbandingan antar kriteria, dan *array* *\$VecWeight* sebagai tempat penyimpanan nilai maksimal yang akan digunakan masing-masing kriteria dari perbandingan vektornya.
2. Baris 3-25: Penentuan nilai vektor perbandingan antar kriteria ke dalam variabel *\$EqVector* sesuai masing-masing kriteria. penentuan nilai vektor tersebut dijelaskan pada baris 6-15. Penentuan nilai variabel *\$VecWeight* sebagai vektor defuzzifikasi diperoleh dengan mencari nilai maksimal dari *array* *\$EqVector* sesuai masing-masing kriteria dijelaskan pada baris 17-23.

4.6.9 Implementasi Normalisasi Vektor defuzzifikasi

Proses normalisasi vektor defuzzifikasi dapat dilakukan setelah memperoleh matriks vektor dan ordinat defuzzifikasi. Implementasi normalisasi vektor defuzzifikasi pada Kode Program 4.9.

```

1  function normal_vector($count,$weight_criteria_rest
2  , $VecWeight,$id_user){
3      $LastWeight = array();
4      $idWeightRes=array();
5      $count_weight=0;
6      $query = "";
7      foreach($weight_criteria_rest as $row ){
8          $idWeightRes[$count_weight]=$row->id;
9          $count_weight++;
10     }
11     for ($i = 0; $i < $count; $i++) {
12         $LastWeight[$i] = $VecWeight[$i]
13         /array_sum($VecWeight);
14         if($count_weight == $count){
15             $query =
16             "UPDATE criteria_weight
17             SET value=".$LastWeight[$i].".
18             WHERE id = ".$idWeightRes[$i].".
19             AND user = '".$id_user."'";
20         }else{
21             $query =
22             "INSERT INTO criteria_weight
23             VALUES('','".$id_user."','".$.
24             $LastWeight[$i].")";
25         }
26     }
27     return array ('LastWeight'=>$LastWeight,
28                 'query'=>$query);
29 }

```

Kode Program 4.9 Normalisasi vektor defuzzifikasi

Penjelasan kode program:

1. Baris 1-2: mendefinisikan inialisasi awal untuk *array \$LastWeight* sebagai tempat penyimpanan vektor defuzzifikasi yang telah di normalisasi yang akan digunakan sebagai bobot kriteria akhir, serta *array \$idWeight* sebagai tempat penyimpanan id untuk penyimpanan bobot kriteria akhir di *database*.
2. Baris 3 : menginisialisasi varabel *\$count_weight* dengan nilai awal 0. Variabel ini digunakan untuk pengecekan dalam penyimpanan ke *database*.
3. Baris 8-21: mendefinisikan perulangan untuk penentuan normalisasi nilai vektor defuzzifikasi atau disebut dengan bobot kriteria akhir. Nilai bobot kriteria akhir dijelaskan dalam kode baris ke 9. Pada baris ke 10-20 menerangkan tentang penyimpanan bobot kriteria akhir ke *database*.

4.6.10 Implementasi Matriks Keputusan

Proses matriks keputusan adalah perubahan data masing-masing kriteria rumah makan dari *database* diubah ke dalam bentuk nilai skala ordinal. Implementasi matriks keputusan pada Kode Program 4.10.

```

1  function matrix($count,$count_res_restaurant,
2      $restaurant,$lat_user,$lng_user){
3      $mKeputusan = array();
4      $mIdRes = array();
5      $mNameRes = array();
6      $count_res = count($count_res_restaurant);
7      $id_res_user =array();
8      $count_res = 0;
9      foreach($count_res_restaurant as $rows ){
10         $id_res_user[$count_res] = $rows->restaurant;
11         $count_res++;
12     }
13     $index = 0;
14     foreach($restaurant as $rows ){
15         for($i=0;$i<$count_res;$i++){
16             if($rows->id == $id_res_user[$i]){
17                 $amount =$rows->amount_menu;
18                 $lat_res =$rows->lat;
19                 $lng_res =$rows->lng;
20                 $rating =$rows->rating;
21                 $price =$rows->average_price;
22                 $open =$rows->open_duration;
23                 $a = pow((sin(deg2rad($lat_res)-
24                     deg2rad($lat_user))/2),2);
25                 $b = cos(deg2rad($lat_res))*
26                     cos(deg2rad($lat_user))*
27                     pow(sin((deg2rad($lng_res)-
28                         deg2rad($lng_user))/2),2);
29                 $distance = 6371.1*2*asin(sqrt($a+$b));
30                 if($amount >=1 && $amount <=5){
31                     $mKeputusan[$index][0] = 1;
32                 }else if($amount >=6 && $amount <=10){
33                     $mKeputusan[$index][0] = 2;
34                 }else if($amount >=11 && $amount <=15){
35                     $mKeputusan[$index][0] = 3;
36                 }else if($amount >=16 && $amount <=20){
37                     $mKeputusan[$index][0] = 4;

```



```

38         }else{
39             $mKeputusan[$index][0] = 5;
40         }
41         if($rating >=1 && $rating <2){
42             $mKeputusan[$index][1] = 1;
43         }else if($rating >=2 && $rating <3){
44             $mKeputusan[$index][1] = 2;
45         }else if($rating >=3 && $rating <4){
46             $mKeputusan[$index][1] = 3;
47         }else if($rating >=4 && $rating <5){
48             $mKeputusan[$index][1] = 4;
49         }else{
50             $mKeputusan[$index][1] = 5;
51         }
52         if($price >40000 && $price <2){
53             $mKeputusan[$index][2] = 1;
54         }else if($price >30000 && $price <= 40000){
55             $mKeputusan[$index][2] = 2;
56         }else if($price >20000 && $price <= 30000){
57             $mKeputusan[$index][2] = 3;
58         }else if($price >10000 && $price <= 20000){
59             $mKeputusan[$index][2] = 4;
60         }else{
61             $mKeputusan[$index][2] = 5;
62         }
63         if($distance >0 && $distance <=3){
64             $mKeputusan[$index][3] = 3;
65         }else if($distance >3 && $distance <=5){
66             $mKeputusan[$index][3] = 2;
67         }else{
68             $mKeputusan[$index][3] = 1;
69         }
70         if($open >0 && $open <=6){
71             $mKeputusan[$index][4] = 1;
72         }else if($open >6 && $open <=12){
73             $mKeputusan[$index][4] = 2;
74         }else if($open >12 && $open <=18){
75             $mKeputusan[$index][4] = 3;
76         }else{
77             $mKeputusan[$index][4] = 4;
78         }
79         $mNameRes[$index]=$rows->name;
80         $mIdRes[$index]=$rows->id;
81         $index++;
82     }
83 }
84 }
85 return array('mKeputusan' =>$mKeputusan,
86             'mNameRes' =>$mNameRes,
87             'mIdRes' =>$mIdRes);
88 }

```

Kode Program 4.10 Matriks keputusan

Penjelasan kode program:

1. Baris 1-3: mendefinisikan inisialisasi awal untuk array *\$mKeputusan* sebagai tempat penyimpanan matriks keputusan, array *\$mIdRes* dan array *\$mNameRes* sebagai tempat penyimpanan *id* dan nama untuk rumah makan dari *database*.
2. Baris 4-6 : menginisialisasi variabel *\$count* dengan nilai banyaknya kriteria, variable *\$count_res* dengan nilai banyaknya rumah makan dan variabel *\$index* dengan nilai awal 0.

- Baris 7-75: mendefinisikan perulangan untuk penentuan normalisasi nilai *array \$mKeputusan*, *\$mIdRes* dan *\$mNameRes*. Baris 9-14 menyimpan data rumah makan ke dalam variabel masing-masing sesuai kriteria. Baris 15-70 penentuan matriks keputusan dengan merubah nilai dari masing-masing kriteria ke dalam bentuk skala ordinal. Baris ke 71-73 menentukan nilai dari *array \$mIdRes*, *\$mNameRes* dan memperbarui variable *\$index*.

4.6.11 Implementasi Normalisasi Matriks Keputusan

Proses normalisasi matriks keputusan dilakukan setelah matriks keputusan dibuat. Implementasi normalisasi matriks keputusan pada Kode Program 4.11.

```

1  function normal_matrix($count,$mKeputusan){
2      $mNKeputusan = array();
3      $sumKeputusan = array();
4      for ($i = 0; $i < $count; $i++) {
5          $sumKeputusan[$i] = 0;
6      }
7      for ($i = 0; $i < $count; $i++) {
8          for ($j=0;$j<count($mKeputusan);$j++){
9              $sumKeputusan[$i] +=
10             ($mKeputusan[$j][$i]*$mKeputusan[$j][$i]);
11          }
12      }
13      for ($i = 0; $i < count($mKeputusan); $i++) {
14          for ($j = 0; $j < $count; $j++) {
15              $mNKeputusan[$i][$j] = $mKeputusan[$i][$j]
16              /sqrt($sumKeputusan[$j]);
17          }
18      }
19      return $mNKeputusan;
20  }
```

Kode Program 4.11 Normalisasi matriks keputusan

Penjelasan kode program:

- Baris 1-2: mendefinisikan inialisasi awal untuk *array \$mNKeputusan* sebagai tempat penyimpanan normalisasi matriks keputusan, *array \$sumKeputusan* sebagai tempat penyimpanan jumlah dari kuadrat nilai matriks keputusan berdasarkan kolom atau kriteria masing-masing.
- Baris 3-5 : menginisialisasi *array sumKeputusan* dengan nilai awal 0.
- Baris 6-10: mendefinisikan perulangan untuk penentuan nilai *array \$sumKeputusan* dengan jumlah kuadrat dari nilai matrik keputusan sesuai dengan kriteria masing-masing.
- Baris 11-16: mendefinisikan perulangan untuk penentuan nilai *array \$mNKeputusan* dengan pembagian nilai *\$mNKeputusan* terhadap akar dari *\$sumKeputusan* sesuai dengan kriteria masing-masing.

4.6.12 Implementasi Normalisasi Terbobot

Proses normalisasi terbobot dilakukan setelah normalisasi matriks keputusan dibuat. Implementasi normalisasi terbobot pada Kode Program 4.12.

```

1  function normal_terbobot($count,$weight_criteria
2      , $cKeputusan, $mNKeputusan) {
```

```

3      $weightCriteriaFahp = array();
4      $mNTKeputusan = array();
5      $count_weight=0;
6      foreach ($weight_criteria as $row) {
7          $weightCriteriaFahp[$count_weight]
8              = $row->value;
9          $count_weight++;
10     }
11     for($i = 0; $i < $cKeputusan; $i++){
12         for ($j = 0; $j < $count; $j++) {
13             $mNTKeputusan[$i][$j] =
14                 $mNKeputusan[$i][$j]*
15                 $weightCriteriaFahp[$j];
16         }
17     }
18     return $mNTKeputusan;
19 }

```

Kode Program 4.12 Normalisasi terbobot

Penjelasan kode program:

1. Baris 1-2: mendefinisikan inisialisasi awal untuk *array* *\$weightCriteriaFahp* sebagai tempat penyimpanan nilai bobot kriteria akhir dari proses FAHP sebelumnya dari *database*, *array* *\$mNTKeputusan* sebagai tempat penyimpanan matriks normalisasi terbobot.
2. Baris 3: menginisialisasi variabel *\$count_weight* dengan nilai awal 0.
3. Baris 4-7: mendefinisikan perulangan untuk penentuan nilai *array* *\$weightCriteriaFahp* dengan dengan nilai bobot akhir dari FAHP sebelumnya yang telah disimpan di *database* serta perbaruan nilai variabel *\$count_weight*.
4. Baris 8-13: mendefinisikan perulangan untuk penentuan nilai *array* *\$mNTKeputusan* dengan perkalian nilai *\$mNKeputusan* terhadap *\$weightCriteriaFahp* sesuai dengan kriteria masing-masing.

4.6.13 Implementasi Solusi Ideal Positif & Ideal Negatif

Proses solusi ideal positif & ideal negatif dilakukan setelah normalisasi terbobot dibuat. Implementasi solusi ideal positif & ideal negatif pada Kode Program 4.13.

```

1      function ideal($count,$cKeputusan,$mNTKeputusan) {
2          $ideal_pos = array();
3          $ideal_neg = array();
4          for ($i = 0; $i < $count; $i++) {
5              $ideal_pos[$i] = 0 ;
6              $ideal_neg[$i] = 100000 ;
7              for ($j = 0; $j < $cKeputusan; $j++) {
8                  if($mNTKeputusan[$j][$i]
9                      > $ideal_pos[$i]){
10                     $ideal_pos[$i] =
11                         $mNTKeputusan[$j][$i];
12                 }
13                 if($mNTKeputusan[$j][$i]
14                     < $ideal_neg[$i]){
15                     $ideal_neg[$i] =
16                         $mNTKeputusan[$j][$i];
17                 }
18             }
19         }
20     }

```

```

18         }
19     }
20     return array("positif" => $ideal_pos,
21                 "negatif" => $ideal_neg);
22 }

```

Kode Program 4.13 Solusi ideal positif & ideal negatif

Penjelasan kode program:

1. Baris 1-2: mendefinisikan inisialisasi awal untuk *array* *\$ideal_pos* dan *\$ideal_neg* sebagai tempat penyimpanan nilai ideal positif dan ideal negatif.
2. Baris 3-15: mendefinisikan perulangan untuk penentuan nilai *array* *\$ideal_pos* dan *\$ideal_neg*. Baris 7-9 menjelaskan penentuan nilai *\$ideal_pos*. Dan baris 10-12 menjelaskan penentuan nilai *\$ideal_neg*.

4.6.14 Implementasi Jarak Solusi Ideal Positif & Jarak Ideal Negatif

Proses jarak solusi ideal positif & jarak ideal negatif dilakukan setelah proses solusi ideal positif & ideal negatif dibuat. implementasi jarak solusi ideal positif & ideal negatif pada Kode Program 4.14.

```

1  function sum_ideal($count,$cKeputusan,
2      $mNTKeputusan,$ideal){
3      $sum_ideal_pos = array();
4      $sum_ideal_neg = array();
5      for ($i = 0; $i < $cKeputusan; $i++) {
6          $sum_ideal_pos[$i] = 0 ;
7          $sum_ideal_neg[$i] = 0 ;
8          for ($j = 0; $j < $count; $j++) {
9              $sum_ideal_pos[$i] +=
10                 (($mNTKeputusan[$i][$j]-
11                     $ideal["positif"][$j])
12                     * ($mNTKeputusan[$i][$j]-
13                         $ideal["positif"][$j]));
14              $sum_ideal_neg[$i] +=
15                 (($mNTKeputusan[$i][$j]-
16                     $ideal["negatif"][$j])
17                     * ($mNTKeputusan[$i][$j]-
18                         $ideal["negatif"][$j]));
19          }
20          $sum_ideal_pos[$i] = sqrt($sum_ideal_pos[$i]);
21          $sum_ideal_neg[$i] = sqrt($sum_ideal_neg[$i]);
22      }
23      return array('positif'=>$sum_ideal_pos,
24                  'negatif'=>$sum_ideal_neg);
25  }

```

Kode Program 4.14 Jarak solusi ideal positif & ideal negatif

Penjelasan kode program:

1. Baris 1-2: mendefinisikan inisialisasi awal untuk *array* *\$sum_ideal_pos* dan *\$sum_ideal_neg* sebagai tempat penyimpanan jarak solusi ideal positif dan negatif.

- Baris 3-14: mendefinisikan perulangan untuk penentuan nilai *array* *\$sum_ideal_pos* dan *\$sum_ideal_neg*.

4.6.15 Implementasi Nilai *Preference*

Proses nilai *preference* dilakukan setelah proses solusi ideal positif & ideal negatif dibuat. Implementasi nilai *preference* pada Kode Program 4.15.

```

1  function preference_value($cKeputusan,$sum_ideal,
2      $id_user,$mIdRes){
3      $PreValue = array();
4      for ($i = 0; $i < $cKeputusan; $i++) {
5          $PreValue[$i] =
6              $sum_ideal['negatif'][$i]
7              / ($sum_ideal['positif'][$i]
8                  + $sum_ideal['negatif'][$i]);
9      }
10     $table_rec = "recommendation";
11     if($type == 1){
12         $table_rec = "alternatif_rec";
13     }
14     $q1 = "DELETE FROM ".$table_rec."
15         WHERE user=".$id_user;
16     $q2 = array();
17     for($i = 0; $i < $cKeputusan; $i++){
18         $q2[$i] = "INSERT INTO ".$table_rec."
19             VALUES('',' ".$id_user."', '".$mIdRes[$i]."',
20                 ".round($PreValue[$i],4).")";
21     }
22     $final = $PreValue;
23     arsort($PreValue, SORT_NUMERIC);
24     return array("PreValue"=>$final,
25                 "asort"=>$PreValue,
26                 "q1"=> $q1,
27                 "q2"=> $q2,);
28 }
29 
```

Kode Program 4.15 Nilai *preference*

Penjelasan kode program:

- Baris 1: mendefinisikan inisialisasi awal untuk *array* *\$PreValue* sebagai tempat penyimpanan nilai *preference*.
- Baris 2-5: mendefinisikan perulangan untuk penentuan nilai *\$PreValue*.
- Baris 6-7: fungsi untuk menghapus data nilai *preference* yang ada di *database* sesuai pengunjung masing-masing.
- Baris 8-13: perulangan fungsi untuk menambahkan data nilai *preference* ke *database* sesuai pengunjung masing-masing.
- Baris 14: fungsi mengurutkan nilai *\$PreValue* dari terbesar ke terkecil.

4.7 Implementasi Antarmuka

Antarmuka sistem terdiri dari halaman menu input data pelanggan, input data rumah makan, FAHP, TOPSIS, hasil rekomendasi dan pengujian. Pada halaman input data pelanggan dapat dilihat pada Gambar 4.21 dan 4.22.

REKOMENDASI RUMAH MAKAN KOTA MALANG MENGGUNAKAN ALGORITMA FAHP-TOPSIS

Penginputan data user Penginputan data rumah makan FAHP TOPSIS Pengujian

Nilai Perbandingan Antar Kriteria

Nama Pelanggan:

Lokasi Pelanggan:

[Search](#) [Simpan](#)

| Kuisiener Perbandingan Kriteria | | | | | | | | | | |
|---------------------------------|-----------------------|---|---|---|---|---|---|---|---|---|
| No | Perbandingan Kriteria | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | Banyak Menu | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 2 | Banyak Menu | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 3 | Banyak Menu | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 4 | Banyak Menu | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 5 | Rating Rumah Makan | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 6 | Rating Rumah Makan | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 7 | Rating Rumah Makan | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 8 | Harga Menu | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 9 | Harga Menu | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 10 | Jarak Rumah Makan | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

Gambar 4.21 Halaman input data pelanggan bagian nama, lokasi, perbandingan kriteria

Rumah makan yang telah user dikunjungi

| No | Rumah Makan | Status | Urutan | Aksi |
|----|------------------------------------|--------|--------|--|
| 1 | Lesehan Bambu & Cafe | | | Add |
| 2 | Ayam Bawang cak Per(suhat) | ✓ | 5 | Edit Delete |
| 3 | Ayam Goreng Nelongso (Suhati) | ✓ | 1 | Edit Delete |
| 4 | Nasi Goreng Mafia | | | Add |
| 5 | Ikana | ✓ | 12 | Edit Delete |
| 6 | Depot Kanton(Oro-oro Dowo, Klojen) | | | Add |
| 7 | Saboten Shokudo,Dieng | | | Add |
| 8 | Ayam Uleg Cak Abit(Sumbersari) | ✓ | 6 | Edit Delete |
| 9 | Rumah Makan Balerang(Suhat) | | | Add |
| 10 | Soto Ambengan | ✓ | 7 | Edit Delete |
| 11 | Pecel Kane(Kerto) | | | Add |
| 12 | Mie Jogging | ✓ | 4 | Edit Delete |
| 13 | Kober Mie Setan Soekarno Hatta | ✓ | 13 | Edit Delete |
| 14 | Warung Biru | | | Add |
| 15 | Rumah makan Rawon Nguling | | | Add |
| 16 | Kimbap Rina | | | Add |
| 17 | Warung Mak Par | | | Add |

Gambar 4.22 Halaman Input Data Pelanggan Bagian Rating

Selanjutnya adalah halaman input data rumah makan dapat dilihat pada Gambar 4.23.

REKOMENDASI RUMAH MAKAN KOTA MALANG MENGGUNAKAN ALGORITMA FAHP-TOPSIS

Penginputan data user Penginputan data rumah makan FAHP TOPSIS Hasil Rekomendasi Pengujian

Input Data Rumah Makan

Import Data Rumah Makan No file chosen

Nama Rumah Makan

Lokasi Rumah Makan

Jumlah Menu Rata-rata Harga

Rating Rumah Makan Lama Buka Rumah Makan (Jam)

Gambar 4.23 Halaman input data rumah makan

Selanjutnya halaman proses FAHP bertujuan mengolah data kriteria pelanggan ke dalam bobot akhir kriteria. Dapat dilihat pada Gambar 4.24 sampai 4.27.

REKOMENDASI RUMAH MAKAN KOTA MALANG MENGGUNAKAN ALGORITMA FAHP-TOPSIS

Penginputan data user Penginputan data rumah makan FAHP TOPSIS Hasil Rekomendasi Pengujian

Proses FAHP

Pengunjung Rumah Makan

| Kriteria | C1 | C2 | C3 | C4 | C5 |
|----------|----|--------|--------|-----|--------|
| C1 | 1 | 0.3333 | 0.3333 | 0.5 | 0.3333 |
| C2 | 3 | 1 | 4 | 3 | 4 |
| C3 | 3 | 0.25 | 1 | 1 | 0.5 |
| C4 | 2 | 0.3333 | 1 | 1 | 2 |
| C5 | 3 | 0.25 | 2 | 0.5 | 1 |
| Jumlah | 12 | 2.1666 | 8.3333 | 6 | 7.8333 |

| Kriteria | C1 | C2 | C3 | C4 | C5 | Jumlah | Bobot Kriteria Awal |
|----------|--------|--------|------|--------|--------|--------|---------------------|
| C1 | 0.0833 | 0.1538 | 0.04 | 0.0833 | 0.0425 | 0.403 | 0.0806 |
| C2 | 0.25 | 0.4616 | 0.48 | 0.5 | 0.5106 | 2.2022 | 0.4404 |
| C3 | 0.25 | 0.1154 | 0.12 | 0.1667 | 0.0638 | 0.7159 | 0.1432 |
| C4 | 0.1667 | 0.1538 | 0.12 | 0.1667 | 0.2553 | 0.8625 | 0.1725 |
| C5 | 0.25 | 0.1154 | 0.24 | 0.0833 | 0.1277 | 0.8164 | 0.1633 |

Gambar 4.24 Halaman proses FAHP bagian matriks kriteria, normalisasi dan bobot kriteria awal

| Vektor Bobot | |
|--------------|--------------|
| Kriteria | Vektor Bobot |
| C1 | 0.4158 |
| C2 | 2.4256 |
| C3 | 0.7493 |
| C4 | 0.9502 |
| C5 | 0.8878 |

| Nilai Prioritas | |
|-----------------|-----------------|
| Kriteria | Nilai Prioritas |
| C1 | 5.1582 |
| C2 | 5.5072 |
| C3 | 5.233 |
| C4 | 5.5087 |
| C5 | 5.4375 |

| Lamda maks | |
|------------|--------|
| | 5.3689 |

| Consistency Index | |
|-------------------|--------|
| | 0.0922 |

| Consistency Ratio (RI = 1.12) | |
|-------------------------------|--------|
| | 0.0823 |

Kesimpulan Konsistensi berdasarkan $CR = 0.0823 < 0.1$, menyatakan matriks diatas konsisten. Sehingga dapat melanjutkan ke langkah berikutnya

Gambar 4.25 Halaman proses FAHP bagian vektor bobot, nilai prioritas hingga cr

| Konversi Matriks Kriteria Perbandingan Berpasangan dengan skala TFN | | | | | | | | | | | | | | | |
|---|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Kriteria | C1 | | | C2 | | | C3 | | | C4 | | | C5 | | |
| | <i>l</i> | <i>m</i> | <i>u</i> | <i>l</i> | <i>m</i> | <i>u</i> | <i>l</i> | <i>m</i> | <i>u</i> | <i>l</i> | <i>m</i> | <i>u</i> | <i>l</i> | <i>m</i> | <i>u</i> |
| C1 | 1 | 1 | 1 | 0.5 | 0.6667 | 1 | 0.5 | 0.6667 | 1 | 0.6667 | 1 | 2 | 0.5 | 0.6667 | 1 |
| C2 | 1 | 1.5 | 2 | 1 | 1 | 1 | 1.5 | 2 | 2.5 | 1 | 1.5 | 2 | 1.5 | 2 | 2.5 |
| C3 | 1 | 1.5 | 2 | 0.4 | 0.5 | 0.6667 | 1 | 1 | 1 | 1 | 1 | 1 | 0.6667 | 1 | 2 |
| C4 | 0.5 | 1 | 1.5 | 0.5 | 0.6667 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.5 | 1 | 1.5 |
| C5 | 1 | 1.5 | 2 | 0.4 | 0.5 | 0.6667 | 0.5 | 1 | 1.5 | 0.6667 | 1 | 2 | 1 | 1 | 1 |

| Jumlah lower, middle dan upper | | | |
|--------------------------------|----------|----------|----------|
| Kriteria | <i>l</i> | <i>m</i> | <i>u</i> |
| C1 | 3.1667 | 4 | 6 |
| C2 | 6 | 8 | 10 |
| C3 | 4.0667 | 5 | 6.6667 |
| C4 | 3.5 | 4.6667 | 6 |
| C5 | 3.5667 | 5 | 7.1667 |
| Jumlah | 20.3 | 26.6667 | 35.8333 |

| Sintesis Fuzzy | | | |
|----------------|-----------|----------|----------|
| Kriteria | <i>Si</i> | | |
| | <i>l</i> | <i>m</i> | <i>u</i> |
| C1 | 0.0884 | 0.15 | 0.2956 |
| C2 | 0.1674 | 0.3 | 0.4926 |
| C3 | 0.1135 | 0.1875 | 0.3284 |
| C4 | 0.0977 | 0.175 | 0.2956 |
| C5 | 0.0995 | 0.1875 | 0.353 |

Gambar 4.26 Halaman proses FAHP bagian konversi TFN dan sintesis fuzzy

| Vektor dan Ordinat Defuzzifikasi | | |
|----------------------------------|-------------|-----------------|
| M2 | M1 | $V(M2 \geq M1)$ |
| K1 | K1 | 1 |
| | K2 | 0.4607 |
| | K3 | 0.8292 |
| | K4 | 0.8878 |
| | K5 | 0.8394 |
| $\sigma'(A1)$ | | 0.4607 |
| K2 | K1 | 1 |
| | K2 | 1 |
| | K3 | 1 |
| | K4 | 1 |
| | K5 | 1 |
| $\sigma'(A2)$ | | 1 |
| K3 | K1 | 1 |
| | K2 | 0.5886 |
| | K3 | 1 |
| | K4 | 1 |
| | K5 | 1 |
| $\sigma'(A3)$ | | 0.5886 |
| K4 | K1 | 1 |
| | K2 | 0.5062 |
| | K3 | 0.9358 |
| | K4 | 1 |
| | K5 | 0.9401 |
| $\sigma'(A4)$ | | 0.5062 |
| K5 | K1 | 1 |
| | K2 | 0.6226 |
| | K3 | 1 |
| | K4 | 1 |
| | K5 | 1 |
| $\sigma'(A5)$ | | 0.6226 |
| Normalisasi Vektor Defuzzifikasi | | |
| Kriteria | Bobot Akhir | |
| C1 | 0.145 | |
| C2 | 0.3147 | |
| C3 | 0.1852 | |
| C4 | 0.1593 | |
| C5 | 0.1959 | |

Gambar 4.27 Halaman proses FAHP bagian vektor dan ordinat defuzzifikasi hingga normalisasi

Halaman proses TOPSIS dilakukan setelah proses FAHP, halaman ini bertujuan mengolah data akhir FAHP berupa bobot kriteria untuk di proses dan menghasilkan rating rumah makan masing-masing pelanggan. Dapat dilihat dapat dilihat pada Gambar 4.28 sampai 4.30.

REKOMENDASI RUMAH MAKAN KOTA MALANG MENGGUNAKAN ALGORITMA FAHP-TOPSIS

Penginputan data user Penginputan data rumah makan FAHP TOPSIS Hasil Rekomendasi Pengujian

Proses TOPSIS

Pengunjung Rumah Makan User 1

| Matriks Keputusan | | | | | |
|-------------------|----|----|----|----|----|
| Alternatif | C1 | C2 | C3 | C4 | C5 |
| A1 | 5 | 4 | 4 | 3 | 3 |
| A6 | 5 | 4 | 2 | 3 | 2 |
| A8 | 3 | 4 | 4 | 2 | 4 |
| A13 | 5 | 4 | 4 | 2 | 2 |
| A14 | 3 | 5 | 4 | 3 | 1 |
| A20 | 4 | 4 | 4 | 3 | 2 |

| Normalisasi Matriks Keputusan | | | | | |
|-------------------------------|--------|--------|--------|--------|--------|
| Alternatif | C1 | C2 | C3 | C4 | C5 |
| A1 | 0.4789 | 0.3904 | 0.4364 | 0.4523 | 0.4867 |
| A6 | 0.4789 | 0.3904 | 0.2182 | 0.4523 | 0.3244 |
| A8 | 0.2873 | 0.3904 | 0.4364 | 0.3015 | 0.6489 |
| A13 | 0.4789 | 0.3904 | 0.4364 | 0.3015 | 0.3244 |
| A14 | 0.2873 | 0.488 | 0.4364 | 0.4523 | 0.1622 |
| A20 | 0.3831 | 0.3904 | 0.4364 | 0.4523 | 0.3244 |

Gambar 4.28 Halaman proses TOPSIS bagian matriks keputusan dan normalisasi

| Normalisasi Terbobot | | | | | |
|----------------------|--------|--------|--------|-------|--------|
| Alternatif | C1 | C2 | C3 | C4 | C5 |
| A1 | 0.0694 | 0.1228 | 0.0808 | 0.072 | 0.0953 |
| A6 | 0.0694 | 0.1228 | 0.0404 | 0.072 | 0.0636 |
| A8 | 0.0417 | 0.1228 | 0.0808 | 0.048 | 0.1271 |
| A13 | 0.0694 | 0.1228 | 0.0808 | 0.048 | 0.0636 |
| A14 | 0.0417 | 0.1535 | 0.0808 | 0.072 | 0.0318 |
| A20 | 0.0555 | 0.1228 | 0.0808 | 0.072 | 0.0636 |

| Solusi Ideal Positif & Ideal Negatif | | | | | |
|--------------------------------------|--------|--------|--------|-------|--------|
| Alternatif | C1 | C2 | C3 | C4 | C5 |
| A+ | 0.0694 | 0.1535 | 0.0808 | 0.072 | 0.1271 |
| A- | 0.0417 | 0.1228 | 0.0404 | 0.048 | 0.0318 |

| Jarak Solusi Ideal Positif & Jarak Ideal Negatif | | |
|--|--------|--------|
| Alternatif | D+ | D- |
| A1 | 0.0442 | 0.0838 |
| A6 | 0.0813 | 0.0486 |
| A8 | 0.0479 | 0.1036 |
| A13 | 0.0746 | 0.0584 |
| A14 | 0.0993 | 0.0562 |
| A20 | 0.0719 | 0.0584 |

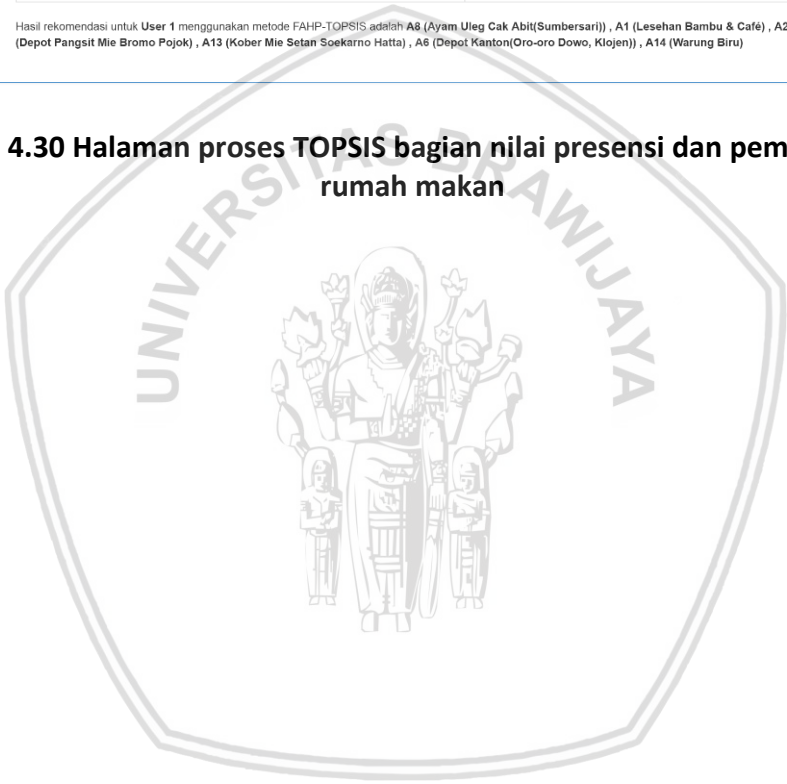
Gambar 4.29 Halaman proses TOPSIS bagian normalisasi terbobot, solusi ideal positif-negatif dan jarak ideal positif-negatif

| Nilai Prefensi | |
|----------------|--------|
| Alternatif | D |
| A1 | 0.6547 |
| A6 | 0.3738 |
| A8 | 0.6839 |
| A13 | 0.4394 |
| A14 | 0.3612 |
| A20 | 0.4481 |

| Sorting Nilai Prefensi | |
|------------------------|--------|
| Alternatif | D |
| A8 | 0.6839 |
| A1 | 0.6547 |
| A20 | 0.4481 |
| A13 | 0.4394 |
| A6 | 0.3738 |
| A14 | 0.3612 |

Hasil rekomendasi untuk User 1 menggunakan metode FAHP-TOPSIS adalah A8 (Ayam Uleg Cak Abit(Sumbersari)) , A1 (Lesehan Bambu & Cafe) , A20 (Depot Pangsit Mie Bromo Pojok) , A13 (Kober Mie Setan Soekarno Hatta) , A6 (Depot Kanton(Oro-oro Dowo, Klojen)) , A14 (Warung Biru)

Gambar 4.30 Halaman proses TOPSIS bagian nilai presensi dan pemeringkatan rumah makan



BAB 5 PENGUJIAN DAN ANALISIS

5.1 Pengujian Korelasi Spearman

Bab ini menjelaskan hasil pengujian terhadap hasil pemeringkatan menggunakan algoritma FAHP-TOPSIS dengan hasil pemeringkatan pelanggan menggunakan uji korelasi Spearman. Pengujian dilakukan terhadap masing-masing pelanggan terhadap rumah makan. Jumlah pelanggan yang diminta pemeringkatan ke-30 rumah makan di kota Malang sebanyak 3 Pelanggan. Rumah makan yang belum pernah dikunjungi tidak akan digunakan sebagai data pengujian. Hal ini dikarenakan rumah makan tersebut belum memiliki peringkat dan tidak bias dikategorikan sebagai peringkat terakhir. Kolom rX adalah peringkat yang dihasilkan sistem dan rY adalah peringkat yang di pilih pakar secara langsung. Kolom d adalah selisih dari nilai rX dengan rY dan kolom D adalah kuadrat dari d . Setelah itu nilai D total secara keseluruhan dan di tentukan nilai p dengan menggunakan Persamaan 2.24. Pengujian pertama dilakukan kepada pelanggan 1. Hasil pengujian dapat dilihat pada Tabel 5.1.

Tabel 5.1 Hasil uji pelanggan pertama

| No | Nama Rumah makan | rX | rY | d | D |
|-------|--|------|------|-------|--------|
| 1 | Ayam D'keprek Kertosentono | 2,5 | 2 | 0,5 | 0,25 |
| 2 | Mie Jogging | 2,5 | 4 | -1,5 | 2,25 |
| 3 | Ayam Bawang cak Per(suhat) | 2,5 | 5 | -2,5 | 6,25 |
| 4 | Kober Mie Setan Soekarno Hatta | 2,5 | 13 | -10,5 | 110,25 |
| 5 | Ayam Goreng Nelongso (Suhat) | 5,5 | 1 | 4,5 | 20,25 |
| 6 | Ayam Uleg Cak Abit(Sumbersari) | 5,5 | 6 | -0,5 | 0,25 |
| 7 | Rumah Makan Dapoer Mie Galau | 7,5 | 10 | -2,5 | 6,25 |
| 8 | Burger Buto 27 | 7,5 | 11 | -3,5 | 12,25 |
| 9 | Soto Ambengan | 9 | 7 | 2 | 4 |
| 10 | Depot Pangsit Mie Bromo Pojok | 10 | 14 | -4 | 16 |
| 11 | Jank Jank Wings, Klojen | 11 | 8 | 3 | 9 |
| 12 | De Chicken, Blimbing | 12,5 | 3 | 9,5 | 90,25 |
| 13 | Ikana | 12,5 | 12 | 0,5 | 0,25 |
| 14 | Waroeng Ayam Bakar Lientang(Sawojajar) | 14 | 9 | 5 | 25 |
| Total | | | | 0 | 302,5 |
| p | | | | | 0,3352 |

Hasil uji korelasi Spearman yang didapat dari Tabel 5.1 adalah 0,3352. Jika diinterpretasikan menggunakan Tabel 2.5 bermakna hubungan moderat. Pengujian Kedua dilakukan kepada pelanggan 2. Hasil pengujian dapat dilihat pada Tabel 5.2.

Tabel 5.2 Hasil uji pelanggan kedua

| No | Nama Rumah makan | rX | rY | d | D |
|-------|--|-----|----|------|---------|
| 1 | Ayam Goreng Nelongso (Suhat) | 1,5 | 1 | 0,5 | 0,25 |
| 2 | Ayam Uleg Cak Abit(Sumbersari) | 1,5 | 11 | -9,5 | 90,25 |
| 3 | Ayam Bakar wong solo(Klojen) | 4,5 | 13 | -8,5 | 72,25 |
| 4 | Lesehan Bambu & Café | 4,5 | 12 | -7,5 | 56,25 |
| 5 | Ayam Bawang cak Per(suhat) | 4,5 | 3 | 1,5 | 2,25 |
| 6 | Bakso President | 4,5 | 9 | -4,5 | 20,25 |
| 7 | Mie Jogging | 9,5 | 6 | 3,5 | 12,25 |
| 8 | Kober Mie Setan Soekarno Hatta | 9,5 | 7 | 2,5 | 6,25 |
| 9 | De Chicken, Blimbing | 9,5 | 10 | -0,5 | 0,25 |
| 10 | Burger Buto 27 | 9,5 | 5 | 4,5 | 20,25 |
| 11 | Warung Mak Par | 9,5 | 2 | 7,5 | 56,25 |
| 12 | Waroeng Ayam Bakar Lientang(Sawojajar) | 9,5 | 8 | 1,5 | 2,25 |
| 13 | Ikana | 13 | 4 | 9 | 81 |
| Total | | | | 0 | 420 |
| p | | | | | -0,1538 |

Hasil uji kerelasi Spearman yang didapat dari Tabel 5.2 adalah -0,1538. Jika diinterpretasikan menggunakan Tabel 2.5 bermakna hubungan kurang berarti. Pengujian Ketiga dilakukan kepada pelanggan 3. Hasil pengujian dapat dilihat pada Tabel 5.3.

Tabel 5.3 Hasil uji pelanggan ketiga

| No | Nama Rumah makan | rX | rY | d | D |
|----|--|----|----|-----|-----|
| 1 | Waroeng Ayam Bakar Lientang(Sawojajar) | 1 | 13 | -12 | 144 |
| 2 | Kedai Pakcik Abin Sulfat | 2 | 14 | -12 | 144 |
| 3 | Ayam dan Bebek Kendang Sawojajar | 3 | 10 | -7 | 49 |
| 4 | Ayam Goreng Nelongso (Suhat) | 4 | 8 | -4 | 16 |
| 5 | Rumah Makan Dapoer Mie Galau | 5 | 9 | -4 | 16 |
| 6 | Lesehan Bambu & Café | 6 | 6 | 0 | 0 |
| 7 | Soto Ambengan | 7 | 11 | -4 | 16 |
| 8 | Burger Buto 27 | 8 | 5 | 3 | 9 |
| 9 | Depot Kanton(Oro-oro Dowo, Klojen) | 9 | 4 | 5 | 25 |

| | | | | | |
|-------|--------------------------------|------|----|------|---------|
| 10 | Saboten Shokudo, Dieng | 10 | 1 | 9 | 81 |
| 11 | Ikana | 11 | 2 | 9 | 81 |
| 12 | Jank Jank Wings, Klojen | 12 | 7 | 5 | 25 |
| 13 | Warung Mak Par | 13 | 12 | 1 | 1 |
| 14 | De Chicken, Blimbing | 14,5 | 3 | 11,5 | 132,3 |
| 15 | Bebek Goreng H Slamet Welirang | 14,5 | 15 | -0,5 | 0,25 |
| Total | | | | 0 | 739,5 |
| p | | | | | -0,3205 |

Hasil uji korelasi Spearman yang didapat dari Tabel 5.3 adalah -0,3205. Jika diinterpretasikan menggunakan Tabel 2.5 bermakna hubungan moderat.

5.2 Analisis Korelasi Spearman

Terdapat 2 penafsiran dalam hasil analisis korelasi dua variabel, yakni kekuatan hubungan dan arah hubungan antar variabel. Kekuatan hubungan dapat ditentukan dengan cara jika hasil korelasi kedua variabel menunjukkan nilai 0 maka kedua variabel tidak memiliki hubungan. Jika hasil korelasi menunjukkan nilai 1 maka kedua variabel memiliki hubungan yang sangat kuat atau sempurna. Jika nilai korelasi mendekati 0 maka hubungan antar variabel semakin melemah begitu pula sebaliknya jika mendekati 1 maka hubungannya semakin kuat.

Penentuan arah hubungan antara kedua variabel dapat dilihat pada nilai dari korelasi yang diperoleh jika bernilai positif menandakan hubungan searah begitupula sebaliknya negatif menandakan hubungan arah terbalik. Hubungan searah di sini adalah jika nilai satu variabel nilainya semakin tinggi maka nilai variabel yang lainnya juga tinggi. Sebaliknya, jika hubungan terbalik maka nilai satu variabel semakin tinggi maka nilai variabel yang lain semakin rendah.

Pada penelitian ini dilakukan 3 kali pengujian korelasi Spearman berdasarkan jumlah partisipan atau pelanggan yang telah melakukan pemeringkatan pakar secara langsung. Pada pengujian pertama didapat nilai p 0,3352 dengan hasil berdasarkan Tabel 2.5 memiliki kedekatan hubungan moderat dengan arah searah. Penelitian kedua didapat nilai p adalah -0,1538 dengan hasil berdasarkan tabel 2.5 memiliki kedekatan hubungan lemah dengan arah berbanding terbalik. Pengujian terakhir, nilai p adalah -0,3205 dengan hasil berdasarkan Tabel 2.5 memiliki kedekatan hubungan moderat dengan arah berbanding terbalik.

Hasil pengujian terdapat 1 nilai positif dengan hubungan moderat searah, 1 nilai negatif hubungan negatif lemah dan 1 nilai negatif dengan hubungan moderat. Hasil yang diharapkan hubungan kedekatan antara pemeringkatan sistem terhadap pelanggan adalah positif. Hal ini dikarenakan kedua variabel yang diujikan merupakan sejenis yakni nilai pemeringkatan, sehingga hubungan arah

yang diharapkan adalah hubungannya searah. Hal ini berbeda dengan jika kedua variabel berbeda jenis, maka hasil nilai p bernilai positif atau tidak akan memiliki kesimpulan yang berbeda. Hal inilah yang menyebabkan hasil pemeringkatan rekomendasi sistem kurang sesuai dengan pemeringkatan rumah makan .

Penyebab kurang sesuainya hasil pemeringkatan sistem terhadap hasil pemeringkatan pelanggan dikarenakan pelanggan yang di sini dipakai sebagai ahli pakar dalam pemeringkatan rumah makan sangat memungkinkan dari faktor di luar dari 5 kriteria yang terdapat dalam sistem. Hal lain yang menyebabkan ketidaksesuaian hasil adalah pemilihan kriteria-kriteria yang digunakan dalam penelitian ini bisa jadi kurang sesuai atau perlu kriteria-kriteria yang lain yang mendasari dalam perekomendasi rumah makan. Diharapkan pada penelitian selanjutnya kriteria-kriteria yang digunakan bisa ditambahkan ataupun diganti dengan kriteria-kriteria yang relevan, serta partisipan yang diikuti sebagai ahli pakar dalam penentuan pemeringkatan mengikuti kriteria-kriteria yang digunakan dalam pengujian.



BAB 6 PENUTUP

Pada bab ini menjelaskan tentang kesimpulan dan saran yang dihasilkan setelah penelitian selesai.

6.1 Kesimpulan

Kesimpulan yang didapat dari hasil penelitian ini dibagi ke dalam beberapa poin yaitu:

1. Penerapan rekomendasi rumah makan menggunakan metode *Fuzzy Analytical Hierarchy Process (FAHP)* dan *Technique For Order Preference By Similarity To Ideal Solution (TOPSIS)* membutuhkan kriteria-kriteria perekomendasian terhadap rumah makan dalam penentuan hasil rekomendasinya. Proses jalannya metode ini terbagi menjadi 2 tahapan: Tahapan Pertama FAHP yakni dimulai matriks perbandingan kriteria, normalisasi matriks perbandingan kriteria, vektor bobot, bobot prioritas, *consistency ratio*, konversi matriks dengan TFN, matriks sintesi *fuzzy*, vektor dan ordinat defuzzifikasi dan normalisasi vektor *fuzzy* atau dikenal sebagai bobot kriteria. Tahap kedua yakni TOPSIS dimulai dari pembuatan matriks keputusan, normalisasi matriks keputusan, matriks normalisasi terbobot, pencarian solusi ideal positif-negatif, pencarian jarak solusi ideal positif-negatif dan nilai *Preference*. Hasil dari nilai *preference* akan dilakukan pengurutan untuk menghasilkan peringkat rumah makan yang direkomendasikan.
2. Pengujian hubungan kedekatan dilakukan dengan menggunakan metode uji korelasi Spearman. Hasil yang didapat dari pemeringkatan oleh sistem terhadap pemeringkatan secara langsung oleh pelanggan pada 3 pengujian adalah 0,3352, -0,1538 dan -0,3205. Hasil inilah menunjukkan hubungan kedekatan yang moderat dan lemah. Selain dari besar nilai uji korelasi juga diperkuat dengan terdapat nilai negatif (-) yang menyatakan hubungan antar kedua variabel perangkingan sistem dengan pemeringkatan pelanggan berbanding terbalik. Penyebab tidak sesuainya hasil pemeringkatan sistem terhadap hasil pemeringkatan pelanggan dikarenakan kriteria yang dipakai pelanggan sebagai ahli pakar dalam pemeringkatan rumah makan sangat memungkinkan dari faktor di luar dari 5 kriteria yang terdapat dalam sistem. Hal lain yang menyebabkan ketidaksesuaian hasil adalah pemilihan kriteria-kriteria yang digunakan dalam penelitian ini bisa jadi kurang sesuai atau perlu kriteria-kriteria yang lain yang mendasari dalam perekomendasian rumah makan.

6.2 Saran

Berdasarkan dari penelitian yang sudah dilakukan, terdapat saran-saran yang diharapkan dapat digunakan pada penelitian selanjutnya yaitu:

1. Kriteria-kriteria yang digunakan dalam perekomendasian rumah makan dapat ditambahkan lagi. Dikarenakan kriteria yang digunakan dalam sistem ini menghasilkan hasil yang kurang sesuai dengan pelanggan.
2. Pelanggan atau pakar yang akan diwawancarai atau dapatkan datanya dari kuesioner hasil pemeringkatannya bisa berhubungan kuat dengan kriteria yang diajukan dalam penelitian.



DAFTAR PUSTAKA

- Chang, D. Y., (1996). Application of the Extent Analysis Method on Fuzzy AHP. *European Journal of Operational Research* 95, 649-655.
- Nurrachman, R., Pinandito, A & Dewi, R.K. 2015 2016, Implementasi Metode AHP-TOPSIS pada sistem Rekomendasi Kuliner. S1. Universitas Brawijaya.
- Pambudi, P.M., Cholissodin, I & Dewi, C. 2016 2017, Implementasi Metode Fuzzy-AHP menggunakan Optimasi Swarm Optimization (PSO) untuk Rekomendasi Pemilihan Tanaman. S1. Universitas Brawijaya.
- Pinandito, A., Ananta, M.T., Brata, K.C, & Fanani, L. 2015. Alternatives weighting in analytic hierarchy process of mobile culinary recommendation system using fuzzy. ResearchGate, [online] Tersedia di: <https://www.researchgate.net/publication/284920900_Alternatives_weighting_in_analytic_hierarchy_process_of_mobile_culinary_recommendation_system_using_fuzzy> [diakses 20 Maret 2017]
- Pinandito, A., Wulandari, C.P & Dewi, R.K. 2015. Culinary Recommendation Systems Using Analytical Hierarchy Process on Google Android Platform. ResearchGate, [online] Tersedia di: <https://www.researchgate.net/publication/292967223_Culinary_Recommendation_Systems_Using_Analytical_Hierarchy_Process_on_Google_Android_Platform> [diakses 20 Maret 2017]
- Pratama, D. & Hansun, S., 2017. Aplikasi Rekomendasi Tempat Makan Menggunakan Algoritma Slope One pada Platform Android. *Indonesian Journal of Computing and Cybernetics Systems*, [online] Tersedia di: <<https://jurnal.ugm.ac.id/ijccs/article/view/15558>> [Diakses 23 Februari 2017]
- Primasari, A. & Siswojo, T., 2012, Promosi Kuliner Lokal sebagai Daya Jual Pariwisata Indonesia untuk Backpacker Asing. S1. Institut Teknologi Bandung.
- Purnomo, E.N.S., Sihwi, S.W & Anggrainingsih, R. 2013. Analisis Perbandingan Menggunakan Metode AHP, TOPSIS, dan AHP-TOPSIS dalam Studi Kasus Sistem Pendukung Keputusan Penerimaan Siswa Program Akselerasi. ITSMART, [online] Tersedia di: <<https://jurnal.uns.ac.id/itsmart/article/view/612>> [diakses 16 Maret 2017]
- Raharjo, Jani dan I Nyoman Sutapa, (2002). Aplikasi Fuzzy Analytical Hierarchy Process dalam Seleksi Karyawan, *Jurnal Teknik Industri*. Vol. 4, no. 2, hal. 82-92.
- Sarwono, J., 2013. Korelasi. Tersedia di <<http://www.jonathansarwono.info/korelasi/korelasi.htm/>> [Diakses 23 Oktober 2018]

- Triana, F.D., 2015. Pengawasan dinas pendapatan daerah kota Malang terkait penagihan pajak restoran di kota Malang (Studi Implementasi Peraturan Daerah Kota Malang Nomor 16 Tahun 2010 Tentang Pajak Daerah) . Tersedia di:<<http://id.portalgaruda.org/?ref=browse&mod=viewarticle&article=326791/>> [Diakses 10 Oktober 2016]
- Yoon, K.P. & Hwang, C.L. 1995. Multiple Attribute Decision Making: An Introduction, Sage Publications, Thousand Oaks, CA.

